



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE14599

**PERANCANGAN KONTROLER PID DAN SIMULATOR
HUMAN MACHINE INTERFACE (HMI) PADA SISTEM
PENGATURAN LEVEL MINYAK UNIT *TEST*
SEPARATOR DI TOTAL E&P INDONESIA**

Mochamad Auliya Rasyid
NRP 2212100200

Dosen Pembimbing
Eka Iskandar, ST., MT.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE14599

***DESIGN PID CONTROLLER AND HUMAN MACHINE
INTERFACE (HMI) SIMULATOR ON THE OIL LEVEL
CONTROL SYSTEM IN THE TEST SEPARATOR AT
TOTAL E&P INDONESIA***

Mochamad Auliya Rasyid
NRP 2212100200

Supervisor
Eka Iskandar, ST., MT.

***ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016***

**PERANCANGAN KONTROLER PID DAN SIMULATOR
HUMAN MACHINE INTERFACE (HMI) PADA SISTEM
PENGATURAN LEVEL MINYAK UNIT TEST SEPARATOR DI
TOTAL E&P INDONESIA**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada**

**Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing



Eka Iskandar, ST., MT.
NIP. 19800528200812 1001



PERANCANGAN KONTROLER PID DAN SIMULATOR HUMAN MACHINE INTERFACE (HMI) PADA SISTEM PENGATURAN LEVEL MINYAK UNIT *TEST SEPARATOR* DI TOTAL E&P INDONESIA

Nama : Mochamad Auliya Rasyid
Pembimbing : Eka Iskandar, ST., MT.

ABSTRAK

Pada *Test Separator* milik Total E&P Indonesia digunakan untuk memisahkan *hydrocarbon* berdasarkan masa jenis kedalam tiga fasa berupa minyak, air dan gas. Level minyak harus dijaga pada ketinggian 50%, karena jika melebihi atau kurang dari *set point* akan mengganggu proses pemisahan serta dapat mengaktifkan *Level Alarm* yang mengakibatkan *shutdown system*. Level diatur menggunakan kontroler *Proportional Integral Derivative* (PID). Namun karena penentuan parameter kontroler menggunakan metode *trial and error*. Respon data dari *real plant* yang didapat memiliki *error steady state*, *overshoot* dan *time constant* yang besar. Beberapa metode kontroler PID dapat digunakan yaitu Ziegler-Nichols, Cohen-Coon dan Analitik, dengan terlebih dahulu mendapatkan model matematis menggunakan identifikasi dinamis bentuk *Auto Regressive Moving Average* (ARMA) melalui metode Pendekatan Penyelesaian Persamaan Linier Simultan Parameter Total (P3LS-PT) dan ditentukan nilai *Integral Absolute Error* (IAE) terkecil dari beberapa model yang didapat. Hasil model yang didapat digunakan juga untuk *training* simulator *Human Machine Interface* (HMI). Hasil simulasi pengujian kontroler, didapatkan bahwa dari metode Analitik $\tau^* = \tau$ dengan parameter $K_p=19.23076$ dan $\tau_i=302.18$ s memiliki respons terbaik dan memenuhi spesifikasi desain dengan 0% *error steady state*, 0% *overshoot* dan *time constant* 300.73 s, respons dengan gangguan memiliki *error steady state* 5% dan memiliki *time constant* 304.73 s, serta sinyal kontrol memiliki nilai fluktuatif terkecil sebesar 10.0323%. Hasil parameter kontroler dan simulator HMI akan menjadi rekomendasi serta media *training* bagi operator di Total E&P Indonesia.

Kata Kunci: ARMA, HMI, IAE, Kontroler PID, Level Minyak, P3LS-PT, Separator, Simulator.

**DESIGN PID CONTROLLER AND HUMAN MACHINE
INTERFACE (HMI) SIMULATOR ON THE OIL LEVEL CONTROL
SYSTEM IN THE TEST SEPARATOR AT TOTAL E&P
INDONESIE**

Name : Mochamad Auliya Rasyid
Supervisor : Eka Iskandar, ST., MT.

ABSTRACT

In Test Separator belongs to Total E & P Indonesie used to separate hydrocarbon-based specify gravity into three phases types such as oil, water and gas. Oil level should be maintained at a height of 50%, because if it exceeds or is less than the set point will disrupt the process of separation and can activate the alarm level that will make shutdown system. The level is set using the controller Proportional Integral Derivative (PID). However, because the determination of the controller parameters using the trial and error method. Response data from real plant that has gained steady state error, the overshoot and the big time constant. Several methods of PID controllers can be used is the Ziegler-Nichols, Cohen-Coon and Analytical, by first obtaining a mathematical model using dynamic identification form Auto Regressive Moving Average (ARMA) through the method of Simultaneous Linear Equations Total Parameter Approach (P3LS-PT) and determined value Integral Absolute Error (IAE), The smallest of the models obtained. Results obtained model was also used for Human Machine Interface (HMI) training simulators. The simulation results of testing the controller, it was found that of the Analytical method $\tau^* = \tau$ parameters $K_p = 19.23076$ and $\tau_i = 302.18$ s has the best response and meet design specifications with 0% error steady state, 0% overshoot and the time constant 300.73 s, the response with disturbance has a 5% error steady state and has a time constant 304.73 s, as well as the control signal having the smallest fluctuation value of 10.0323%. Results parameter controller and HMI simulator will be a recommendation as well as media training for operators at Total E & P Indonesie.

Key Word: ARMA, HMI, IAE, Oil Level, P3LS-PT, PID Controllers, Separator, Simulator.

KATA PENGANTAR

Puji dan syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan buku Tugas Akhir ini. Shawatul serta salam senantiasa tercurah pula kepada nabi besar baginda Rasulullah Muhammad SAW.

Buku Tugas Akhir ini disusun untuk melengkapi salah satu syarat memperoleh gelar sarjana teknik di jurusan Teknik Elektro ITS. Buku yang berjudul **“Perancangan Kontroler PID dan Simulator *Human Machine Interface* (HMI) pada Sistem Pengaturan Level Minyak Unit Test Separator di Total E&P Indonesia”** dipersembahkan juga untuk kemajuan riset dan teknologi Indonesia khususnya untuk ITS, Fakultas Teknologi Industri, Jurusan Teknik Elektro, dan bidang studi Teknik Sistem Pengaturan serta bagi perusahaan yang diteliti yaitu Total E&P Indonesia.

Hambatan dan rintangan selalu ada dalam pengerjaan. Namun dukungan dan bantuan terus mengalir hingga penulis dapat menyelesaikan buku ini. Pada kesempatan ini tak lupa penulis menyampaikan rasa terima kasih kepada beberapa pihak, antara lain:

1. Orang tua dan adik tercinta yang secara tidak langsung menjadi sumber semangat dalam menyelesaikan buku ini.
2. Dosen Pembimbing, Bapak Eka Iskandar, ST., MT. atas bimbingannya selama mengerjakan Tugas Akhir ini.
3. Pembimbing Lapangan, Bapak Noorjahid, Bapak Fanzzy Anom dan Bapak Eka Ramadanur Putra atas bimbingannya selama mengerjakan Tugas Akhir ini di Total E&P Indonesia.
4. Teman-teman angkatan 2012 Teknik Elektro ITS, khususnya bidang studi Teknik Sistem Pengaturan.
5. Semua pihak yang turut membantu dalam pengerjaan Tugas Akhir ini yang tidak dapat disebutkan satu per satu.

Penulis menyadari dan memohon maaf karena masih banyak kekurangan pada Tugas Akhir ini. Kritik dan saran selalu penulis nantikan agar menjadi lebih baik pada masa mendatang. Akhir kata, penulis berharap Tugas Akhir ini dapat bermanfaat dan menjadi acuan dalam penelitian selanjutnya.

Surabaya, Januari 2016

Penulis

TABLE OF CONTENT

COVER	i
AUTHENTICITY	iii
APPROVAL SHEET	v
ABSTRACT	vii
PREFACE	xi
TABLE OF CONTENT	xiii
ILLUSTRATION	xv
TABLE LIST	xix
 CHAPTER I INTRODUCTION	 1
1.1 Background	1
1.2 Problem Formulation	2
1.3 Problem Constraint	2
1.4 Research Objective	2
1.5 Writing Systematic	3
1.6 Relevance	4
 CHAPTER II LITERATURE OBSERVATION	 5
2.1 Test Separator	5
2.1.1 Test Separator Components	7
2.2 Piping & Instrumentation Diagram (P&ID)	9
2.2.1 Connection Process Lines	10
2.2.2 General Instrument	10
2.2.3 Actuator and Final Element Standard	11
2.2.4 Identification Letter	12
2.3 System Identification	13
2.3.1 Auto Regressive Moving Average (ARMA)	13
2.3.2 Simultaneous Linear Equations Total Parameter	14
2.3.3 Z Transform	14
2.3.4 Laplace Transform	15
2.4 Model Validation	15
2.5 Controller PID	15
2.5.1 P Controller	16
2.5.2 I Controller	16
2.5.3 D Controller	17
2.5.4 PI Controller	17
2.5.5 PD Controller	18

2.5.6	PID Controller	18
2.6	Design PID Controller Ziegler-Nichols Method.....	19
2.7	Design PID Controller Cohen-Coon Method	20
2.8	Design PID Controller Analytical Method	22
BAB III SYSTEM DESIGN		23
3.1	Human Machine Interface Design	23
3.1.1	Human Machine Interface	23
3.1.1.1	Header Part.....	24
3.1.1.2	Body Part.....	24
3.1.1.3	Footer Part.....	25
3.1.2	Human Machine Interface Window	25
3.1.2.1	Home Window	25
3.1.2.2	Faceplate Window.....	26
3.1.2.3	Trend Window.....	27
3.1.2.4	Process Overview Window	28
3.1.2.5	System Identification Window	29
3.2	Design PID Controller Level Control System	23
3.3	Control System Architechure	25
3.3.1	Level Control System P&ID	26
3.4	System Identification	27
3.4.1	Closed Loop System Identification	27
3.4.2	Closed Loop Model Validation	29
3.4.3	Open Loop System Identification.....	30
3.5	Design Specification.....	32
3.6	Design PID Controller.....	32
3.6.1	PID Controller Ziegler-Nichols Method.....	32
3.6.2	PID Controller Cohen-Coon Method	32
3.6.3	PID Controller Analytical Method	32
CHAPTER IV TESTING AND ANALYSIS		41
4.1	Human Machine Interface Testing	41
4.1.1	Call Window Menu Testing	41
4.1.2	Control Menu Testing Testing.....	41
4.1.3	Disturbance Menu Testing	42
4.1.4	Alarm Status Testing.....	43
4.1.5	System Identification Menu Testing.....	44
4.1.6	Indicator HMI Testing.....	44
4.2	Model Testing.....	48

4.2.1	Closed Loop Testing.....	48
4.2.2	Open Loop Testing	49
4.3	Testing PID Controller.....	51
4.3.1	Controller Ziegler-Nichols Method	51
4.3.1.1	P Controller.....	51
4.3.1.2	PI Controller	53
4.3.1.3	PID Controller	54
4.3.2	PID Controller Cohen-Coon Method.....	56
4.3.2.1	P Controller.....	56
4.3.2.2	PI Controller	58
4.3.2.3	PID Controller	59
4.3.3	PID Controller Analytical Method	61
4.3.3.1	PI Controller $\tau^* = 5\tau$	61
4.3.3.2	PI Controller $\tau^* = 2\tau$	63
4.3.3.3	PI Controller $\tau^* = \tau$	64
4.3.3.4	PI Controller $\tau^* = 0.8\tau$	65
4.3.3.5	PI Controller $\tau^* = 0.6\tau$	67
4.4	Testing PID Controller with Disturbance.....	69
4.4.1	PI Controller $\tau^* = \tau$ with Disturbance	69
4.4.2	PI Controller $\tau^* = 0.8\tau$ with Disturbance	70
4.4.3	PI Controller $\tau^* = 0.6\tau$ with Disturbance	72
CHAPTER V CLOSING.....		77
5.1	Conclusion	77
5.2	Suggestion.....	78
BIBLIOGRAPHY		79
ENCLOSURE.....		83
BIOGRAPHY		102

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN	iii
HALAMAN PENGESAHAN	v
ABSTRAK	vii
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Sistematika Penulisan	3
1.6 Relevansi	4
BAB II TINJAUAN PUSTAKA	5
2.1 Test Separator	5
2.1.1 Komponen Test Separator	7
2.2 Piping & Instrumentation Diagram (P&ID)	9
2.2.1 Standar Garis Penghubung	10
2.2.2 Standar Instrumen Umum	10
2.2.3 Standar Aktuator dan Elemen Proses	11
2.2.4 Huruf Identifikasi Element P&ID	12
2.3 Identifikasi Sistem	13
2.3.1 Auto Regressive Moving Average (ARMA)	13
2.3.2 Pendekatan Penyelesaian Persamaan Linier Simultan Parameter Total	14
2.3.3 Transformasi Z	14
2.3.4 Transformasi Laplace	15
2.4 Validasi Model	15
2.5 Kontroler PID	15
2.5.1 Kontroler P	16
2.5.2 Kontroler I	16
2.5.3 Kontroler D	17
2.5.4 Kontroler PI	17

2.5.5	Kontroler PD	18
2.5.6	Kontroler PID.....	18
2.6	Perancangan Kontroler PID Metode Ziegler-Nichols.....	19
2.7	Perancangan Kontroler PID Metode Cohen-Coon.....	20
2.8	Perancangan Kontroler PI Metode Analitik.....	22
BAB III PERANCANGAN SISTEM		23
3.1	Perancangan <i>Human Machine Interface</i>	23
3.1.1	Tampilan Desain <i>Human Machine Interface</i>	23
3.1.1.1	Bagian <i>Header</i>	24
3.1.1.2	Bagian <i>Body</i>	24
3.1.1.3	Bagian <i>Footer</i>	25
3.1.2	<i>Window Human Machine Interface</i>	25
3.1.2.1	<i>Window Home</i>	25
3.1.2.2	<i>Window Faceplate</i>	26
3.1.2.3	<i>Window Trend</i>	27
3.1.2.4	<i>Window Process Overview</i>	28
3.1.2.5	<i>Window System Identification</i>	29
3.2	Perancangan Kontroler PID Sistem Pengaturan Level	29
3.3	Arsitektur Sistem Pengaturan	31
3.3.1	P&ID Sistem Pengaturan Level.....	32
3.4	Identifikasi Sistem	33
3.4.1	Identifikasi Sistem <i>Closed Loop</i>	33
3.4.2	Validasi Model Sistem <i>Closed Loop</i>	35
3.4.3	Identifikasi Sistem <i>Open Loop</i>	37
3.5	Spesifikasi Desain.....	38
3.6	Perancangan Kontroler PID.....	39
3.6.1	Kontroler PID Metode Ziegler-Nichols.....	39
3.6.2	Kontroler PID Metode Cohen-Coon	39
3.6.3	Kontroler PID Metode Analitik.....	39
BAB IV PENGUJIAN DAN ANALISIS.....		41
4.1	Pengujian <i>Human Machine Interface</i> (HMI).....	41
4.1.1	Uji <i>Call Window Menu</i>	41
4.1.2	Uji <i>Control Menu</i>	41
4.1.3	Uji <i>Disturbance Menu</i>	42
4.1.4	Uji <i>Alarm Status</i>	43
4.1.5	Uji <i>System Identification Menu</i>	44
4.1.6	Uji Indikator HMI	44

4.2	Pengujian Model	48
4.2.1	Uji <i>Closed Loop</i>	48
4.2.2	Uji <i>Open Loop</i>	49
4.3	Pengujian Kontroler PID	51
4.3.1	Kontroler Metode Ziegler-Nichols	51
4.3.1.1	Kontroler Tipe P	51
4.3.1.2	Kontroler Tipe PI	53
4.3.1.3	Kontroler Tipe PID	54
4.3.2	Kontroler Metode Cohen-Coon	56
4.3.2.1	Kontroler Tipe P	56
4.3.2.2	Kontroler Tipe PI	58
4.3.2.3	Kontroler Tipe PID	59
4.3.3	Kontroler Metode Analitik	61
4.3.3.1	Kontroler Tipe PI dengan $\tau^* = 5\tau$	61
4.3.3.2	Kontroler Tipe PI dengan $\tau^* = 2\tau$	63
4.3.3.3	Kontroler Tipe PI dengan $\tau^* = \tau$	64
4.3.3.4	Kontroler Tipe PI dengan $\tau^* = 0.8\tau$	65
4.3.3.5	Kontroler Tipe PI dengan $\tau^* = 0.6\tau$	67
4.4	Pengujian Kontroler PID dengan Gangguan	69
4.4.1	Kontroler Tipe PI $\tau^* = \tau$ dengan Gangguan	69
4.4.2	Kontroler Tipe PI $\tau^* = 0.8\tau$ dengan Gangguan	70
4.4.3	Kontroler Tipe PI $\tau^* = 0.6\tau$ dengan Gangguan	72
BAB V PENUTUP		77
5.1	Kesimpulan	77
5.2	Saran	78
DAFTAR PUSTAKA		79
LAMPIRAN		83
RIWAYAT PENULIS		101

DAFTAR GAMBAR

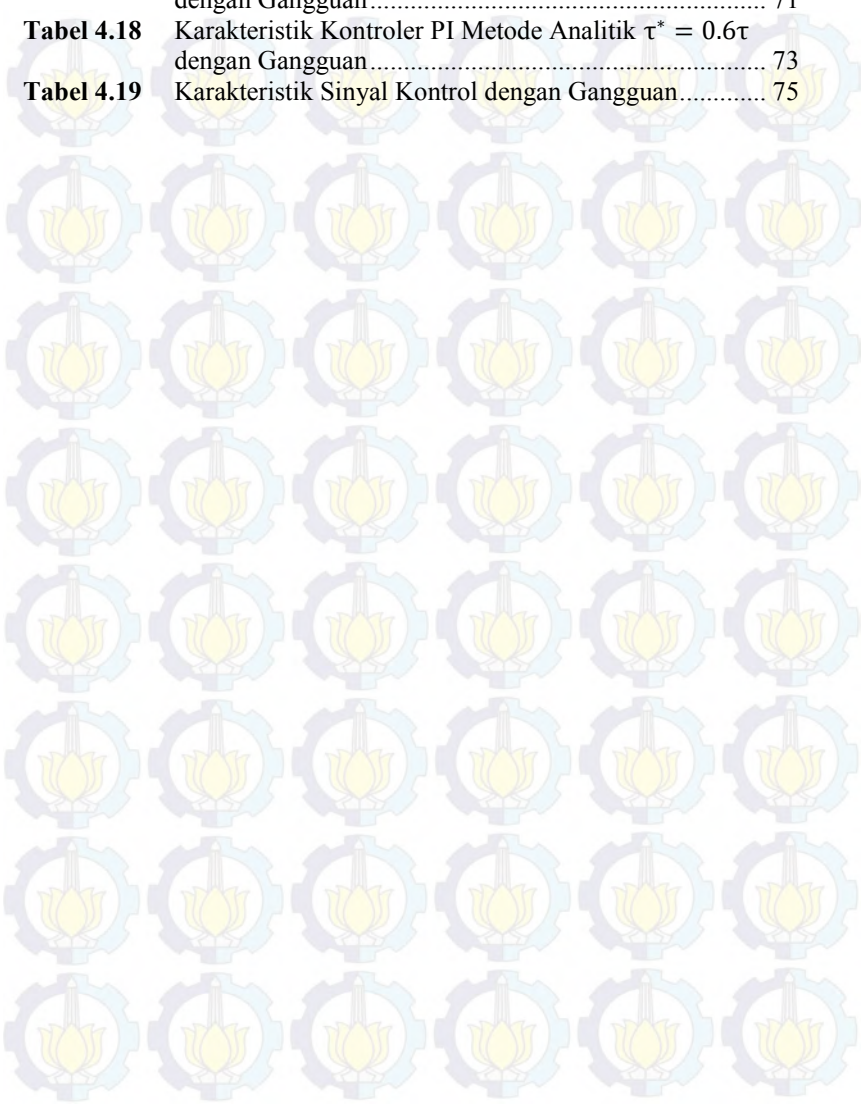
Gambar 2.1	<i>Test Separator</i>	5
Gambar 2.2	<i>Diverter Plate/ Deflector</i>	7
Gambar 2.3	<i>Demister/ Mist Extractor</i>	8
Gambar 2.4	<i>Vortex Breaker</i>	8
Gambar 2.5	Blok Diagram Bentuk ARMA	13
Gambar 2.6	Respons Kurva S Metode Ziegler-Nichols	20
Gambar 2.7	Respons Kurva Cohen-Coon	21
Gambar 3.1	Tampilan Desain HMI	23
Gambar 3.2	Tampilan Bagian <i>Header</i>	24
Gambar 3.3	Tampilan Bagian <i>Body</i>	24
Gambar 3.4	Tampilan Bagian <i>Footer</i>	25
Gambar 3.5	Tampilan <i>Window Home</i>	26
Gambar 3.6	Tampilan <i>Window Faceplate</i>	27
Gambar 3.7	Tampilan <i>Window Trend</i>	28
Gambar 3.8	Tampilan <i>Window Process Overview</i>	28
Gambar 3.9	Tampilan <i>Window System Identification</i>	29
Gambar 3.10	Diagram Alur Metode Penelitian	31
Gambar 3.11	Arsitektur Sistem Pengaturan Level Minyak	32
Gambar 3.12	P&ID Sistem Pengaturan Level Minyak Test Separator	32
Gambar 3.13	Identifikasi Sistem	33
Gambar 3.14	Respons Hasil Model Estimasi	36
Gambar 3.15	Blok Diagram Sistem <i>Closed Loop</i>	37
Gambar 3.16	Respons Step <i>Open Loop</i>	38
Gambar 4.1	Pengujian <i>Control</i> dan <i>Automatic Mode</i>	42
Gambar 4.2	Pengujian <i>Disturbance Menu Non Active</i> dan <i>Active</i> ..	42
Gambar 4.3	Pengujian <i>Alarm Status</i>	43
Gambar 4.4	Pengujian <i>System Identification</i>	44
Gambar 4.5	Pengujian Indikator <i>Window Home</i>	45
Gambar 4.6	Pengujian Indikator <i>Window Faceplate</i>	46
Gambar 4.7	Pengujian Indikator <i>Window Trend</i>	46
Gambar 4.8	Pengujian Indikator <i>Window Process Overview</i>	47
Gambar 4.9	Pengujian Indikator <i>Window System Identification</i>	47
Gambar 4.10	Respons Sistem <i>Closed Loop</i>	48
Gambar 4.11	Respons Sistem <i>Open Loop</i>	50
Gambar 4.12	Respons Kontroler Tipe P Metode Ziegler-Nichols	52
Gambar 4.13	Respons Kontroler Tipe PI Metode Ziegler-Nichols ...	53

Gambar 4.14	Respons Kontroler Tipe PID Metode Ziegler-Nichols	54
Gambar 4.15	Respons Kontroler Metode Ziegler-Nichols.....	56
Gambar 4.16	Respons Kontroler Tipe P Metode Cohen-Coon	57
Gambar 4.17	Respons Kontroler Tipe PI Metode Cohen-Coon.....	58
Gambar 4.18	Respons Kontroler Tipe PID Metode Cohen-Coon.....	59
Gambar 4.19	Respons Kontroler Metode Cohen-Coon	61
Gambar 4.20	Respons Kontroler Tipe PI dengan $\tau^* = 5\tau$	62
Gambar 4.21	Respons Kontroler Tipe PI dengan $\tau^* = 2\tau$	63
Gambar 4.22	Respons Kontroler Tipe PI dengan $\tau^* = \tau$	64
Gambar 4.23	Respons Kontroler Tipe PI dengan $\tau^* = 0.8\tau$	66
Gambar 4.24	Respons Kontroler Tipe PI dengan $\tau^* = 0.6\tau$	67
Gambar 4.25	Respons Kontroler Tipe PI Analitik	68
Gambar 4.26	Respons Kontroler Tipe PI $\tau^* = \tau$ dengan Gangguan.	69
Gambar 4.27	Respons Kontroler Tipe PI $\tau^* = 0.8\tau$ dengan Gangguan	71
Gambar 4.28	Respons Kontroler Tipe PI $\tau^* = 0.6\tau$ dengan Gangguan	72
Gambar 4.29	Respons Kontroler Tipe PI dengan Gangguan	74
Gambar 4.30	Respons Sinyal Kontrol Tipe PI dengan Gangguan	74

DAFTAR TABEL

Tabel 2.1	<i>Data Sheet Test Separator</i>	6
Tabel 2.2	<i>Standard Connection Process Lines</i>	9
Tabel 2.3	<i>Standard General Instruments</i>	10
Tabel 2.4	<i>Standard Element and Actuator</i>	11
Tabel 2.5	<i>Standard Identification Letter</i>	12
Tabel 2.6	Efek Perubahan Parameter Kontroler PID	19
Tabel 2.7	Parameter PID Metode Ziegler-Nichols	20
Tabel 2.8	Parameter PID Metode Cohen-Coon	21
Tabel 3.1	Keterangan Warna HMI	25
Tabel 3.2	Keterangan Warna Grafik Trend	27
Tabel 3.3	<i>Instrumentation Code List</i>	33
Tabel 3.4	Validasi Model	36
Tabel 3.5	Hasil Parameter Kontroler PID Ziegler-Nichols	39
Tabel 3.6	Hasil Parameter Kontroler PID Cohen-Coon	39
Tabel 3.7	Hasil Parameter Kontroler PI Analitik	40
Tabel 4.1	Pengujian <i>Call Window Menu</i>	41
Tabel 4.2	Pengujian Indikator HMI	44
Tabel 4.3	Karakteristik Sistem <i>Closed Loop</i>	49
Tabel 4.4	Karakteristik Sistem <i>Open Loop</i>	50
Tabel 4.5	Karakteristik Kontroler P Metode Ziegler-Nichols	41
Tabel 4.6	Karakteristik Kontroler PI Metode Ziegler-Nichols	53
Tabel 4.7	Karakteristik Kontroler PID Metode Ziegler-Nichols	55
Tabel 4.8	Karakteristik Kontroler P Metode Cohen-Coon	57
Tabel 4.9	Karakteristik Kontroler PI Metode Cohen-Coon	58
Tabel 4.10	Karakteristik Kontroler PID Metode Cohen-Coon	60
Tabel 4.11	Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 5\tau$	62
Tabel 4.12	Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 2\tau$	63
Tabel 4.13	Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = \tau$	65
Tabel 4.14	Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 0.8\tau$	66
Tabel 4.15	Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 0.6\tau$	67
Tabel 4.16	Karakteristik Kontroler PI Metode Analitik $\tau^* = \tau$ dengan Gangguan	70

Tabel 4.17	Karakteristik Kontroler PI Metode Analitik $\tau^* = 0.8\tau$ dengan Gangguan.....	71
Tabel 4.18	Karakteristik Kontroler PI Metode Analitik $\tau^* = 0.6\tau$ dengan Gangguan.....	73
Tabel 4.19	Karakteristik Sinyal Kontrol dengan Gangguan.....	75



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam industri energi khususnya perminyakan. Minyak dan gas diperoleh dari sumur, namun karena masing-masing sumur memiliki karakteristik yang berbeda-beda [1]. Maka setiap sumur membutuhkan pengujian terlebih dahulu sebelum memulai produksi berupa *historical* dan *data logging*, hal tersebut yang digunakan untuk mengetahui karakteristik dan informasi dalam produksi sumur seperti umur, jangka waktu, potensi dan hal lain dalam proses produksi [2].

Unit *test separator* tiga fasa merupakan salah satu alat yang digunakan untuk pengujian dan monitoring sumur minyak dan gas milik Total E&P Indonesia. Komponen tersebut akan memisahkan fluida dari dalam sumur kedalam tiga fasa, yaitu minyak, air dan gas berdasarkan perbedaan masa jenis. Dan setiap fasa tersebut akan keluar ke *outlet* masing masing. Untuk menjaga setiap fasa keluar ke outlet masing-masing dibutuhkan sistem pengaturan berbagai *variable* seperti level air dan minyak, suhu, dan tekanan agar mendapatkan hasil proses yang akurat [3].

Salah satu *variable* yang dikontrol dalam *test separator* tiga fasa adalah level minyak. Dengan cara membandingkan level yang dibaca oleh sensor dengan *set point* yang diinginkan. Dengan cara mengatur bukaan *control valve* pada *outlet* keluaran level minyak [4]. Jenis kontroler yang digunakan pada Total E&P Indonesia adalah jenis *Proportional Integral Derivative* (PID). Namun, dalam penentuan parameter kontroler tersebut masih menggunakan metode *trial and error* sehingga mengakibatkan respons yang didapat terdapat *error steady state*, *overshoot* dan waktu *transient* yang besar. Hal tersebut dapat menurunkan tingkat akurasi proses pengujian sumur serta dapat membahayakan proses jika mengaktifkan alarm. Metode *trial and error* digunakan karena belum adanya model matematis dan media *training simulator* operator dari *test separator* yang digunakan [5].

Berbagai macam metode dalam menentukan parameter PID, seperti secara analitik, Ziegler-Nichols, Cohen-Coon dan masih banyak lagi. Dari beberapa metode yang ada terdapat kelebihan dan kekurangan pada hasil respons yang didapat. Maka dari itu dibutuhkan identifikasi model sistem pengaturan level minyak agar dapat menentukan parameter

kontroler sehingga dapat melakukan penalaan ulang dalam menentukan parameter PID sehingga hasil respons yang didapat 0% *error steady state*, 0% *maximum overshoot* dan *time constant* sistem hasil perancangan tidak lebih dari dari *time constant* sistem *real*, dengan membandingkan beberapa metode penalaan.

1.2 Perumusan Masalah

Permasalahan yang menjadi topik penelitian ini adalah bagaimana menjaga level minyak sebesar 50% dalam *test separator*. Karena jika level minyak terlalu tinggi melebihi 50%, minyak dapat masuk kesaluran gas dan dapat mengaktifkan *level alarm high-high*. Serta jika level minyak terlalu rendah, gas dapat masuk kesaluran minyak dan dapat mengaktifkan *level alarm low-low*. Jika terdapat fasa yang tidak keluar sesuai dengan *outlet* akan mengganggu proses dan aktifnya *level alarm* dapat mengakibatkan *shutdown system*. Sistem didesain dengan spesifikasi desain 0% *error steady state*, 0% *maximum overshoot* dan *time constant* sistem hasil perancangan tidak lebih dari dari *time constant* sistem *real*. Menggunakan beberapa metode penalaan kontroler PID.

1.3 Batasan Masalah

Dalam mengerjakan penelitian ini, dari ketiga fasa dalam plant hanya minyak saja yang dibahas dengan *input* sistem menggunakan sinyal step yang berubah. Data didapatkan dari *Distributed Control System* (DCS) yang ada pada *test separator* tiga fasa di Sisi Nubi Field pada 14 Agustus 2015 mulai 12.48 WITA sampai 15.49 WITA. Kontroler yang digunakan menggunakan tipe PID sesuai yang dapat digunakan di lapangan.

1.4 Tujuan Penelitian

Penelitian ini bertujuan mendapatkan parameter kontroler PID yang dapat menjaga level minyak sebesar 50% dalam *test separator* dengan spesifikasi desain 0% *error steady state*, 0% *maximum overshoot* dan *time constant* sistem hasil perancangan tidak lebih dari dari *time constant* sistem *real*. Identifikasi plant menggunakan bentuk *Auto Regressive Moving Average* (ARMA) dengan metode Pendekatan Penyelesaian Persamaan Linier Simultan-Parameter Total (P3LS-PT). dengan validasi *error* model metode *Integral Absolute Error* (IAE). Dan untuk metode penalaan kontroler PID secara Ziegler Nichols, Cohen

Coon dan analitik. Hasil dari setiap respons kontroler akan dianalisis karakteristiknya sehingga mendapatkan respons sesuai dengan spesifikasi desain yang telah ditentukan serta merancang simulator *Human Machine Interface* (HMI) dari *test separator* yang dapat digunakan sebagai media pelatihan bagi operator lapangan.

1.5 Sistematika Penulisan

Sistematika penulisan dalam penelitian ini terdiri dari lima bab yaitu pendahuluan, teori penunjang, metodologi penelitian, pengujian dan analisis, penutup.

BAB I PENDAHULUAN

Pada bab ini, dijelaskan tentang latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, sistematika penulisan, dan relevansi.

BAB II TEORI PENUNJANG

Dasar pemikiran dan pengetahuan dari sistem yang akan diidentifikasi seperti Cara kerja, komponen dan P&ID *Test Separator* Tiga Fasa, metode identifikasi sistem dinamik, validasi model berdasarkan *error*, dan perancangan kontroler.

BAB III METODOLOGI PENELITIAN

Metodologi menjelaskan mengenai langkah-langkah untuk mencapai tujuan dan kesimpulan dari penelitian. Hasil dari bagian ini adalah model yang memiliki *error* terkecil dan parameter kontroler PID yang cocok untuk disimulasikan dan hasil perancangan HMI.

BAB IV PENGUJIAN DAN ANALISIS

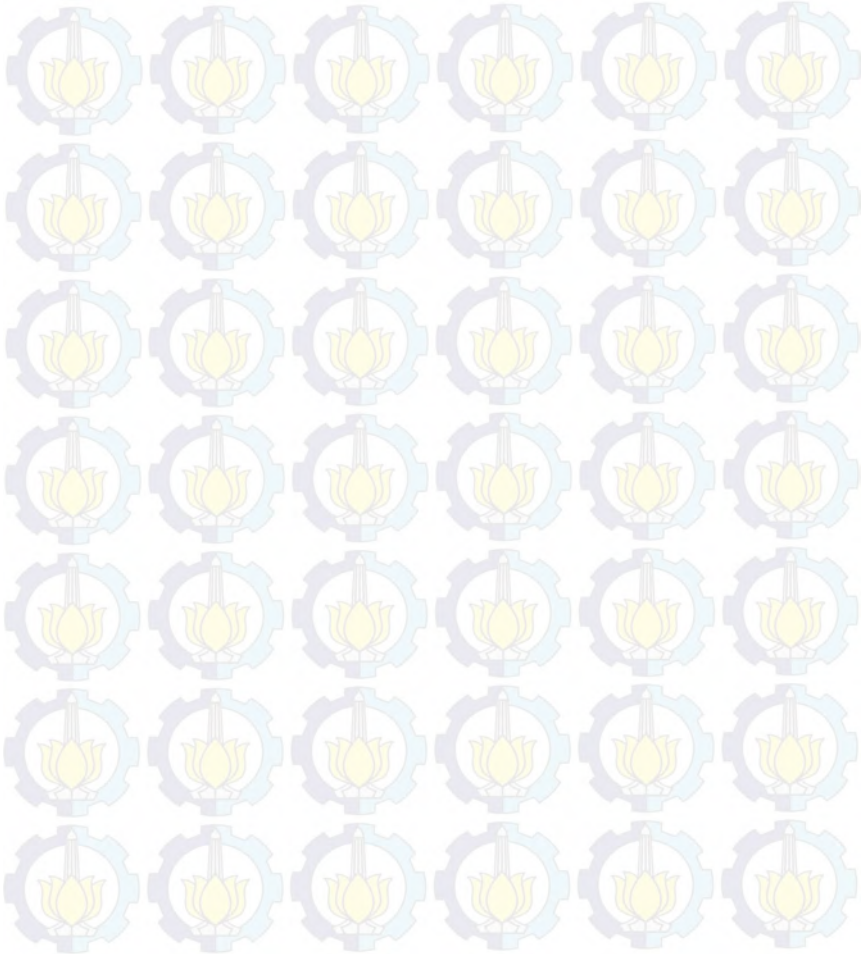
Merupakan hasil pengujian dan analisis sistem secara simulasi dari tiap metode perancangan kontroler PID serta pengujian HMI yang telah dibuat.

BAB V PENUTUP

Pada bab ini, diuraikan tentang kesimpulan akhir dari penelitian ini dan saran-saran mengenai kemungkinan pengembangan penelitian ini.

1.6 Relevansi

Model dari plant *test separator* Tiga Fasa yang valid dapat menjadi acuan sebagai penentu parameter kontroler PID yang cocok. Dan dapat digunakan oleh perusahaan Total E&P Indonesia sebagai rekomendasi dan pembandingan pada *real plant* serta simulator HMI dapat digunakan untuk media pelatihan bagi operator lapangan.

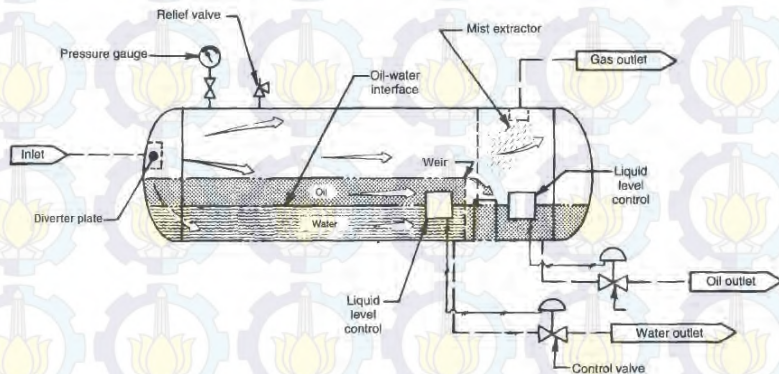


BAB II

TEORI PENUNJANG

Pada Bab ini akan dibahas mengenai materi dasar dalam penyusunan Tugas Akhir. Beberapa hal yang dibahas meliputi tinjauan pustaka mengenai penelitian yang berkaitan seperti *test separator*, identifikasi sistem dan kontroler.

2.1 *Test Separator* [6,7,8,9]



Gambar 2.1 *Test Separator* [6]

Test separator merupakan sebuah *vessel* besar yang digunakan untuk memisahkan fluida dari dalam sumur berupa *hydrocarbon* kedalam beberapa fasa seperti yang dapat dilihat pada Gambar 2.1. Pada *test separator* yang dimiliki oleh Total E&P Indonesia bertipe tiga fasa, yang artinya hasil pemisahan dari *test separator* tersebut akan dipisahkan menjadi tiga fasa dalam bentuk air, minyak dan gas. Fluida masuk kedalam *vessel* melalui *inlet* dan akan menabrak dinding berupa *deflector* atau *diverter plate*. *Deflector* tersebut berguna untuk memecah fluida dari sumur. Hasil pemecahan dari *deflector* akan jatuh kebagian bawah *vessel*. Pemisahan terjadi secara natural berdasarkan perbedaan masa jenis. Air dengan masa jenis paling besar akan berada dibagian bawah, minyak dengan masa jenis lebih ringan akan berada diatas air. Dan gas dengan masa jenis paling ringan akan berada paling atas.

Didalam separator terdapat sekat yang disebut *weir* yang digunakan untuk menjadi pembatas antara *compartement* minyak dan air. Fluida, pertama akan masuk ke *compartement* air, minyak yang berada diatasnya akan tumpah ke *compartement* minyak. Dan gas akan keluar melalui *gas outlet* melalui saluran bagian atas *vessel*. Pada saluran sebelum *gas outlet* terdapat *demister* yang berfungsi agar gas yang masuk ke saluran gas benar benar kering. Pada setiap *outlet liquid* yaitu minyak dan gas terdapat *vortex breaker* yang berguna untuk mengurangi turbulensi akibat keluarnya liquid dari dalam *vessel* melalui *water outlet* dan *oil outlet* [7].

Tabel 2.1 *Data Sheet Test Separator* [8]

No	Item		Keterangan
1	Name Service		Test Separator
2	Code		32-VZ-1101
3	Vessel	Height (mm)	6000
4		Diameter (mm)	2200
5	Height Weir (mm)		750
6	Pressure	Design (barg)	87
		Working (barg)	55
7	Temperature	Design (°C)	110
		Working (°C)	95
8	Oil Level Alarm	LAHH (mm)	1400
		LAH (mm)	1250
		LAL (mm)	500
		LALL (mm)	300
9	Interface Level Alarm	ILAH (mm)	700
		ILAL (mm)	500
		ILALL (mm)	300

Dalam menjalankan proses pada *test separator*, dibutuhkan pengendalian agar proses berjalan dengan baik dan aman. Seperti level minyak dan air serta *pressure* gas. Level air diatur sesuai *set point*, karena jika level air terlalu rendah maka minyak dapat masuk ke saluran air dan jika air terlalu tinggi melebihi ketinggian *weir*, air dapat masuk ke bagian minyak. Level minyak diatur sesuai *set point*, karena jika level minyak terlalu rendah maka gas dapat masuk ke saluran minyak serta

jika level minyak terlalu tinggi, minyak dapat masuk ke saluran gas. *Pressure* gas diatur sesuai dengan *set point* Karena jika *pressure* terlalu tinggi atau terlalu rendah dapat membahayakan *vessel*. Setiap fasa harus keluar sesuai dengan salurannya masing-masing karena jika terdapat anomali dapat mengganggu dan membahayakan proses [9]. Beberapa spesifikasi *separator* yang digunakan dapat dilihat pada Tabel 2.1.

2.1.1 Komponen *Test Separator* [10,11]

Didalam *vessel* separator terdapat beberapa komponen yang menunjang proses pemisahan fluida sebagai berikut:

1. *Inlet*

Inlet merupakan saluran tempat masuknya fluida ke dalam *vessel*.

2. *Outlet*

Outlet merupakan saluran tempat keluaranya fluida yang telah terpisahkan dari *vessel* yaitu berupa *gas outlet*, *oil outlet* dan *water outlet*.

3. *Diverter plate / deflector*

Diverter plate atau *deflector* merupakan *plate* yang terletak didalam tepat berhadapan dengan *inlet* yang berfungsi untuk memecah aliran yang masuk. Fluida dari *inlet* berupa *liquid* akan jatuh kebagian bawah *vessel* dan gas akan tetap lewat seperti pada Gambar 2.2.



Gambar 2.2 *Diverter Plate/ Deflector* [10]

4. *Weir*

Weir adalah sebuah *plate* atau bendungan yang berfungsi untuk memisahkan minyak dari air, dan mencegah air masuk ke *compartement* minyak.

5. *Demister/ Mist extractor*

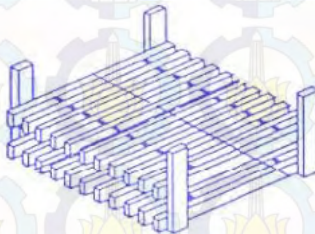
Berfungsi untuk mencegah *liquid* yang terikut dalam gas masuk ke saluran gas, dengan terlebih dahulu disaring oleh *demister* diharapkan fasa yang keluar melalui *outlet* gas berupa gas seperti pada Gambar 2.3.



Gambar 2.3 *Demister/ Mist Extractor* [10]

6. *Vortex breaker*

Vortex breaker berfungsi untuk mencegah pusaran pada outlet liquid seperti pada Gambar 2.4.



Gambar 2.4 *Vortex Breaker* [10]

7. *Pressure Indicator*

Pressure Indicator berfungsi sebagai indikator tekanan dalam *vessel* biasanya berupa *pressure gauge* atau dapat berupa *relieve valve*.

8. *Level Indicator*

Level indicator berfungsi sebagai indikator level *liquid* dalam *vessel* yang dapat berupa *level transmitter* atau *level glass*.

9. *Level Control Valve*

Level control valve adalah *valve* yang berfungsi untuk mengatur ketinggian *liquid* dalam *vessel* dengan cara mengatur aliran pada *outlet*.



10. *Pressure Control Valve*

Pressure control valve adalah *valve* yang berfungsi untuk mengatur tekanan dalam *vessel* dengan cara mengatur aliran pada *gas outlet*.





2.2 *Piping & Instrumentation Diagram (P&ID) [12,13]*

Bentuk dan simbol standar untuk skema pada *process control* sama dengan yang berlaku pada skema rangkaian elektronik. Pada *process control* skema tersebut biasa disebut *piping and instrumentation drawing (P&ID)*. Semua simbol dan artinya telah diatur oleh *American National Standard Institute (ANSI)* dan *The Instrumentation, System and Automation (ISA)*. Semua standard tersebut tertuang dalam peraturan *Instrumentation Symbol and Identification, ANSI/ISA 85.1-1984 (R1992)*. Pada peraturan ini, berbagai macam aturan untuk tiap komponen dalam suatu skema P&ID dijelaskan,

Tabel 2.2 *Standard Connection Process Lines [13]*

No	Jenis	Gambar
1	<i>Instrument Supply or Connection to Process</i>	
2	<i>Pneumatic Signal</i>	

Tabel 2.2 *Standard Connection Process Lines* (lanjutan) [13]

No	Jenis	Gambar
3	<i>Electric Signal</i>	
4	<i>Hydraulic Signal</i>	
5	<i>Capillary Tube</i>	
6	<i>Data Link</i>	







2.2.1 Standard Garis Penghubung (*Connection Process Lines*)

Sebuah standar yang menjelaskan tentang tipe garis. Standar ini diperlukan untuk merepresentasikan berbagai macam koneksi dalam plant seperti pada Tabel 2.2.







2.2.2 Standar Instrumen Umum (*General Instrument*)

Standar ini mendefinisikan simbol-simbol yang biasa dipakai dalam berbagai macam instrumen yang dibutuhkan untuk pengukuran dan pengaturan sistem yang ada di plant. Instrument tersebut termasuk sensor, *transmitter* dan kontroler seperti pada Tabel 2.3.

Tabel 2.3 *Standard General Instruments* [13]

<i>Name</i>	<i>Primary Location</i>	<i>Field Mounted</i>	<i>Auxiliary Location</i>
<i>Discrete Instruments</i>			
<i>Shared Display, Shared Control</i>			



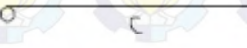

Tabel 2.3 *Standard General Instruments* (lanjutan) [13]

<i>Name</i>	<i>Primary Location</i>	<i>Field Mounted</i>	<i>Auxiliary Location</i>
<i>Computer Function</i>			
<i>Programmable Logic Controller</i>			


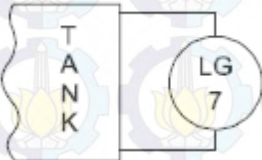
2.2.3 Standard Aktuator dan Elemen Proses

Simbol aktuator dan elemen menjadi salah satu komponen penting dalam proses. Antara lain yaitu *control valve*, *conveyor*, motor dan lainnya seperti pada Tabel 2.4.

Tabel 2.4 *Standard Element and Actuator* [13]

No	Nama	Gambar
1.	<i>Valve</i>	
2.	<i>Orifice</i>	
3.	Konveyor	
4.	Pompa	

Tabel 2.4 *Standard Element and Actuator* (lanjutan) [13]

No	Nama	Gambar
5.	Silinder	
6.	<i>Tank with Level Glass</i>	

2.2.4 Huruf Identifikasi Element P&ID (*Identification Letter*)

Pada umumnya simbol pada skema P&ID disertai oleh kombinasi huruf dan angka. Kombinasi tersebut menunjukkan fungsi dan tujuan instrument tersebut dalam suatu plant sepertipada Tabel 2.5.

Tabel 2.5 *Standard Identification Letter* [13]

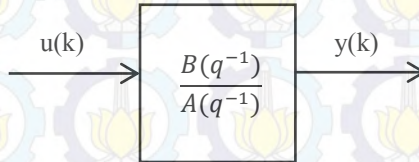
<i>Letter</i>	<i>First Letter</i>	<i>Second Letter</i>
A	<i>Analysis</i>	<i>Alarm</i>
C	<i>Conductivity</i>	<i>Control</i>
D	<i>Density</i>	
E	<i>Voltage</i>	<i>Primary Element</i>
F	<i>Flow</i>	
G	<i>Gaging</i>	<i>Glass</i>
H	<i>Hand</i>	<i>High</i>
I	<i>Current</i>	<i>Indicate</i>
L	<i>Level</i>	<i>Low</i>
P	<i>Pressure</i>	<i>Point</i>
T	<i>Temprature</i>	<i>Transmit</i>
V	<i>Viscosity</i>	<i>Valve</i>
Y		<i>Relay</i>
Z	<i>Position</i>	<i>Actuator</i>

2.3 Identifikasi Sistem [14,15,16]

Identifikasi sistem digunakan untuk mengetahui model dari suatu sistem, dan yang akan dibahas meliputi identifikasi dinamis bentuk *Auto Regressive Moving Average* (ARMA) metode parameter total, transformasi domain waktu diskrit ke domain Z, dan tranformasi domain Z ke domain Laplace.

2.3.1 Auto Regressive Moving Average (ARMA)

Dalam mendapatkan model matematis dari plant terdapat banyak metode, salah satunya dengan metode dinamik dengan cara membandingkan *input* dan *output* sehingga mendapatkan model dari *black box* seperti diagram blok pada Gambar 2.5.



Gambar 2.5 Blok Diagram Bentuk ARMA [14]

Salah satu metode identifikasi sistem dinamik yaitu dengan bentuk ARMA atau *auto regressive moving average*, model ini akan membentuk model matematis dengan menghasilkan model dalam domain diskrit dengan memiliki bagian *numerator* seperti pada Persamaan 1.1 dan bagian *denominator* seperti pada Persamaan 1.2. Sehingga bentuk ARMA memiliki bentuk seperti Persamaan 1.3.

$$A(q^{-1}) = a_0 + a_1 q^{-1} + \dots + a_{nA} q^{-nA} \quad (1.1)$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_{nB} q^{-nB} \quad (1.2)$$

$$\left(1 + \sum_{i=1}^{nA} a_i q^{-i}\right) y(k) = \left(\sum_{i=1}^{nB} b_i q^{-i}\right) u(k) \quad (1.3)$$

Dengan n adalah orde hasil estimasi sistem, a dan b koefisien sistem, $y(k)$ *output* dan $u(k)$ *input*.

2.3.2 Pendekatan Penyelesaian Persamaan Linier Simultan Parameter Total (P3LS-PT) [15].

Salah satu metode untuk mendapatkan parameter model dapat ditentukan dengan metode Pendekatan Penyelesaian Persamaan Linier Simultan Parameter Total (P3LS-PT). Jika diketahui suatu sistem model bentuk ARMA memiliki bentuk seperti pada Persamaan 1.4, dengan θ sebagai koefisien hasil estimasi dan \emptyset data perubahan *input* dan *output* dapat ditentukan nilai koefisiennya seperti pada Persamaan 1.5.

$$\begin{bmatrix} -y(m-1) \dots y(m-nA) + x(m) + x(m-1) \dots x(m-nB) \\ \vdots \\ -y(m+n-1) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} y(n) \\ \vdots \\ y(m+n) \end{bmatrix} \quad (1.4)$$

$\emptyset \qquad \qquad \qquad Y$

(1.5)

2.3.3 Transformasi Z [16].

Data yang didapat dari model ARMA ini berada pada domain waktu diskrit, agar mudah dalam melakukan permodelan maka model tersebut di transformasikan ke domain Z terlebih dahulu menggunakan Persamaan 1.6 sehingga akan didapatkan model dalam domain Z.

$$\begin{aligned} X(z) &= Z[x(k)] \\ &= \sum_{k=1}^{\infty} x(k) z^{-k} \end{aligned} \quad (1.6)$$

2.3.4 Transformasi Laplace

Untuk memudahkan perancangan kontroler, model dari bentuk z diubah ke bentuk Laplace dengan cara mengubah *variable* z dengan Persamaan 1.7, dengan memasukkan nilai T sebagai *time sampling* sistem sesuai dengan metode konversi dengan metode *forward*.

$$z = 1 + Ts \quad (1.7)$$

2.4 Validasi Model [17]

Validasi model digunakan untuk mendapatkan model identifikasi yang mendekati model sebenarnya. Model matematika yang didapat perlu divalidasi agar dapat dilihat seberapa tepat dengan sistem asli pada kondisi riil. Salah satunya menggunakan metode validasi *Integral Absolute Error* (IAE).

IAE adalah pengukuran akurasi pada nilai deret waktu secara statistik, dengan memutlakan terlebih dahulu nilai *error* baru kemudian diintegrasikan. Kriteria IAE ini telah digunakan secara luas karena kemudahan perhitungan *integral* baik secara analitis maupun eksperimental. Formulasi dasar IAE dapat dilihat pada Persamaan 1.8.

$$\int_0^{\infty} |e_i| \quad (1.8)$$

Dimana batas atas ∞ dapat diganti dengan n jumlah data sebanyak yang diambil dan i adalah urutan data. Sedangkan e adalah kesalahan data hasil dari permodelan terhadap hasil pengukuran seperti yang dirumuskan pada Persamaan 1.9.

$$\begin{aligned} e_i &= y_A - y_M \\ i &= 1, 2, 3, \dots, n \end{aligned} \quad (1.9)$$

dimana y_A adalah nilai data asli hasil pengukuran, dan y_M adalah nilai data hasil permodelan.

2.5 Kontroler PID [18,19,20]

Pengaturan PID tersusun atas tiga aksi yaitu pengaturan secara proporsional, pengaturan secara integral, dan pengaturan secara derivatif. Dengan masukan berupa sinyal *error* dan keluarannya berupa

sinyal kontrol. Secara umum algoritma PID dapat didefinisikan sebagai berikut.

2.5.1 Kontroler P (*Proportional Controller*)

Kontroler tipe P merupakan tipe kontroler yang sinyal masukannya proporsi dengan keluarannya. Kontroler jenis ini dapat dinyatakan dalam Persamaan 1.10.

$$u(t) = K_p e(t) \quad (1.10)$$

Atau dalam fungsi alih dapat ditulis seperti pada Persamaan 1.11.

$$\frac{U(s)}{E(s)} = K_p \quad (1.11)$$

Selain dalam bentuk , kontroler proportional juga biasa disebut dalam bentuk *proportional band* seperti pada Persamaan 1.12.

$$K_p = \frac{100\%}{PB} \quad (1.12)$$

$$PB = \frac{100\%}{K_p}$$

2.5.2 Kontroler I (*Integral Controller*)

Kontroler tipe I memiliki karakteristik seperti halnya sebuah integral. Keluaran *controller* sangat dipengaruhi oleh penjumlahan sinyal *error* berdasarkan waktu. Keluaran *controller* ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Apabila sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. *Transfer function* dari unit *control integral* seperti pada persamaan 13.

$$u(t) = \frac{1}{T_i} K_p \int e(t) dt \quad (1.13)$$

Atau dalam fungsi alih dapat ditulis seperti pada Persamaan 1.14.

$$\frac{U(s)}{E(s)} = \frac{K_p}{T_i s} \quad (1.14)$$

2.5.3 Kontroler D (*Derivative Controller*)

Kontroler Tipe D memiliki sifat seperti halnya suatu operasi derivatif, yaitu mengurangi sinyal *error* berdasarkan waktu. Perubahan yang mendadak pada masukan controller, akan mengakibatkan perubahan yang sangat besar dan cepat. Kontroler jenis ini dapat dinyatakan seperti pada Persamaan 1.15.

$$u(t) = K_p T_d \frac{de(t)}{dt} \quad (1.15)$$

Atau dalam fungsi alih dapat ditulis seperti pada Persamaan 1.16.

$$\frac{U(s)}{E(s)} = K_p T_d s \quad (1.16)$$

2.5.4 Kontroler PI (*Proportional Integral Controller*)

Kontroler jenis ini memiliki aksi gabungan dari kontroler tipe P dan tipe I dan dinyatakan dalam Persamaan 1.17.

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt \right] \quad (1.17)$$

Atau dalam fungsi alih dapat ditulis seperti pada Persamaan 1.18.

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} \right) \quad (1.18)$$

2.5.5 Kontroler PD (*Proportional Derivative Controller*)

Kontroler jenis ini memiliki aksi gabungan dari kontroler tipe P dan tipe D dan dinyatakan dalam Persamaan 1.19

$$u(t) = K_p \left[e(t) + T_d \frac{de(t)}{dt} \right] \quad (1.19)$$

Atau dalam fungsi alih dapat ditulis seperti pada persamaan 1.20

$$\frac{U(s)}{E(s)} = K_p (1 + T_d s) \quad (1.20)$$

2.5.6 Kontroler PID (*Proportional Integral Derivative Controller*)

Kontroler jenis ini memiliki aksi gabungan dari kontroler tipe P, tipe I dan tipe D, dinyatakan dalam Persamaan 1.21

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (1.21)$$

Atau dalam fungsi alih dapat ditulis seperti pada Persamaan 1.22

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (1.22)$$

Secara umum kontroler PID yang terdiri dari nilai T memiliki fungsi yang berbeda beda jika terjadi penambahan maupun pengurangan nilai, perubahan nilai parameter tersebut akan berdampak kepada beberapa karakteristik respons seperti *Rise Time* (T_r), *Overshoot*, *Settling Time* (T_s) dan kestabilan. Efek perubahan tersebut dapat dilihat seperti pada Tabel 2.6.

Tabel 2.6 Efek Perubahan Parameter Kontroler PID

Perlakuan Parameter	Rise Time (Tr)	Overshoot	Settling Time (Ts)	Kestabilan
Pembesaran K_p	Berkurang	Meningkat	Sedikit bertambah	Menurun
Pembesaran K_i / Pengecilan τ_i	Berkurang	Meningkat	Bertambah	Menurun
Pembesaran K_d / Pembesaran τ_d	Berubah Sedikit	Berkurang	Berkurang	Meningkat

2.6 Perancangan Kontroler PID Metode Ziegler-Nichols [21,22]

Metode Ziegler-Nichols merupakan metode penalaan kontroler PID yang umum digunakan pada industri. Metode ini menentukan nilai parameter PID berdasarkan respons plant terhadap masukan sinyal step secara eksperimental. Terdapat dua metode dalam metode ini, yaitu metode kurva S dan osilasi. Dalam tugas akhir ini akan dibahas menggunakan metode kurva.

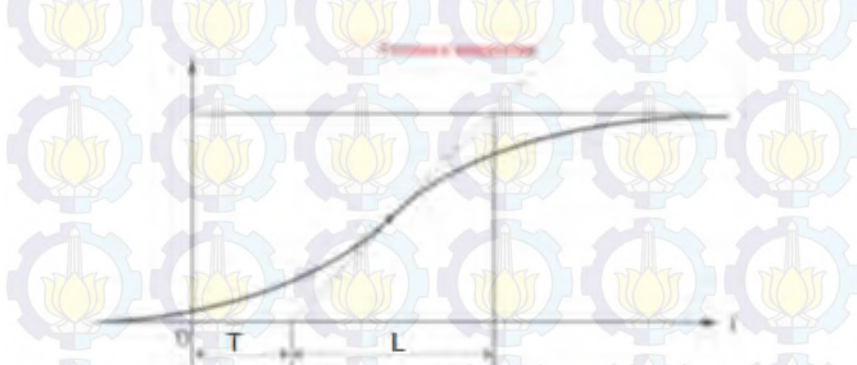
Penalaan *controller* PID dengan metode kurva reaksi Ziegler-Nichols akan mendapatkan fungsi transfer seperti pada Persamaan 1.22.

$$G_c = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

$$G_c = 1.2 \frac{T}{L} \left(1 + \frac{1}{2Ls} + 0.5Ls \right) \quad (1.22)$$

$$G_c = 0.6T \frac{\left(s + \frac{1}{L} \right)^2}{s}$$

Metode ini didapatkan dengan melihat respons sistem dengan memberikan sinyal step. Respons yang didapat akan membentuk kurva S. Kurva berbentuk S dikarakteristikan oleh dua parameter yaitu waktu tunda L dan konstanta waktu tunda T. Konstanta waktu tunda T ditentukan dengan menggambarkan garis singgung dengan sumbu waktu dan garis respons. Dengan mencari variable L dan T yang nantinya akan dimasukkan kedalam tabel PID. Nilai L dan T didapatkan dengan mengikuti seperti yang ada pada Gambar 2.6



Gambar 2.6 Respons Kurva S Metode Ziegler-Nichols [21]

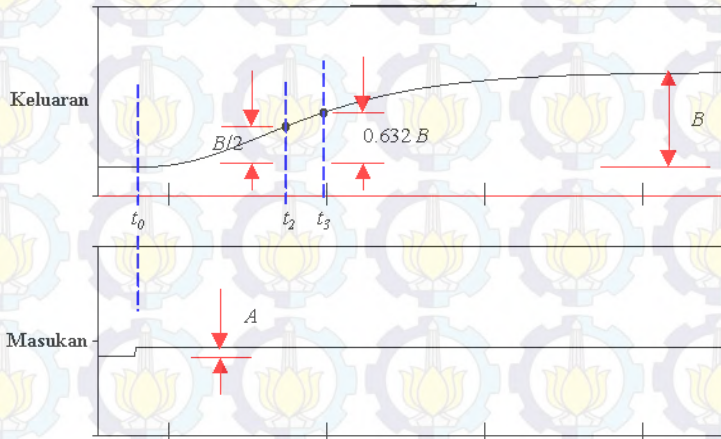
Setelah mendapatkan nilai sesuai dengan kebutuhan, nilai L dan T dimasukkan kedalam tabel sehingga mendapatkan parameter kontroler PID seperti pada Tabel 2.6.

Tabel 2.7 Parameter PID Metode Ziegler-Nichols [21]

Type Kontroler	Kp	Ti	Td
P	$\frac{T}{L}$	-	-
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	-
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

2.7 Perancangan Kontroler PID Metode Cohen-Coon [17]

Pada metode ini respons sistem dibagi menjadi tiga kriteria waktu yaitu, t_0 adalah waktu ketika masukan mulai diberikan pada *plant*, t_2 adalah waktu ketika keluaran sistem mencapai setengah dari nilai keadaan tunaknya, dan t_3 adalah waktu yang dicapai ketika keluaran sistem mencapai 63,2% dari nilai keadaan tunaknya. Pada Gambar 2.7 memperlihatkan pembagian waktu pada respons sistem sebagai acuan untuk menentukan parameter penalaan kontroler PID.



Gambar 2.7 Respons Metode Cohen-Coon

Pada Tabel 2.8 memperlihatkan aturan penalaan parameter kontroler PID dengan metode Cohen-Coon. Besarnya nilai P, I, dan D ditentukan berdasarkan nilai dari K , r , dan τ_{del} .

Tabel 2.8 Parameter PID Metode Cohen-Coon [17]

Tipe Kontroler	Kp	Ti	Td
P	$\frac{1}{K.r} (1 + \frac{r}{3})$	-	-

Tabel 2.8 Parameter PID Metode Cohen-Coon (lanjutan) [17]

Tipe Kontroler	Kp	Ti	Td
PI	$\frac{1}{K.r} (0,9 + \frac{r}{12})$	$\tau_{del} \frac{30 + 3r}{9 + 20r}$	-
PID	$\frac{1}{K.r} (\frac{4}{3} + \frac{r}{4})$	$\tau_{del} \frac{32 + 6r}{13 + 8r}$	$\tau_{del} \frac{4}{11 + 2r}$

$$\begin{aligned}
 t_1 &= \frac{t_2 - \ln(2)t_3}{1 - \ln(2)} \\
 \tau &= t_3 - t_1 \\
 \tau_{DEL} &= t_1 - t_0
 \end{aligned} \tag{1.23}$$

$$\begin{aligned}
 K &= \frac{B}{A} \\
 r &= \frac{\tau_{del}}{\tau}
 \end{aligned}$$

2.8 Perancangan Kontroler PI Metode Analitik [23]

Kontroler PI dengan metode analitik dirancang karena dapat menentukan spesifikasi performa respons yang diinginkan. Dengan menentukan spesifikasi desain yang diinginkan dapat ditentukan parameter kontroler yang cocok yaitu seperti pada Persamaan 1.24 dan Persamaan 1.25.

$$\tau = \tau_i \tag{1.24}$$

$$K_p = \frac{\tau_i}{\tau^* K} \tag{1.25}$$

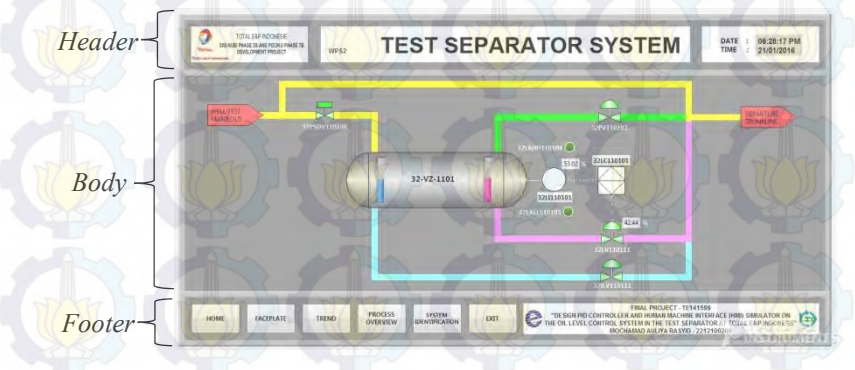
Pada Bab ini dibahas mengenai metodologi penelitian yang digunakan berupa perancangan *Human Machine Interface* (HMI), perancangan sistem pengaturan level minyak, identifikasi sistem, validasi model dan perancangan kontroler PID dari beberapa metode yang telah ditentukan sebelumnya.

3.1 Perancangan *Human Machine Interface* (HMI)

Perancang HMI menggunakan *software LabVIEW* dari National Instruments. Dengan membuat program pada bagian *block diagram* dan tampilan HMI pada bagian *front panel*. Tampilan Desain HMI terdiri dari tiga bagian utama yaitu *header*, *body* dan *footer* serta terdiri dari lima *window*, yaitu *Home*, *Faceplate*, *Trend*, *Process Overview*, dan *System Identification*.

3.1.1 Tampilan Desain HMI

Tampilan desain HMI memiliki dimensi *pixels* dengan *height* sebesar 665 dan *width* 1272. Memiliki tiga bagian utama yaitu *header*, *body* dan *footer* ditunjukkan seperti yang dapat dilihat pada Gambar 3.1. Tampilan dibuat semirip mungkin dengan HMI pada kondisi *real* agar memudahkan operator menggunakan simulator HMI untuk keperluan mempelajari sistem pada *test separator*.



Gambar 3.1 Tampilan Desain HMI

3.1.1.1 *Bagian Header*

Bagian *header* merupakan bagian paling atas dari HMI memiliki dimensi *pixels* dengan *height* sebesar 108 dan *width* 1272. Terdiri dari beberapa bagian seperti pada Gambar 3.2. Dan berisikan keterangan umum *window*.

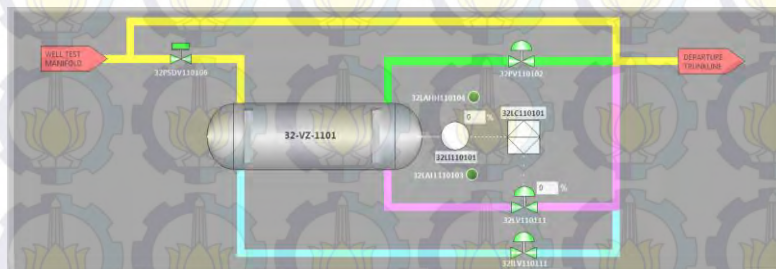


Gambar 3.2 Tampilan Bagian *Header*

Bagian *header* terdiri dari *logo*, *title* dan *time*. Pada bagian *logo* terdapat logo perusahaan Total E&P Indonesia dan keterangan lokasi *plant* berada pada Sisi-Nubi *Offshore Field*. Pada bagian *title* berisi judul dari isi tiap masing masing *window* dan pada bagian *time* berisi waktu saat HMI *running* dengan format waktu jam: menit: detik dan tanggal/ bulan/ tahun.

3.1.1.2 *Bagian Body*

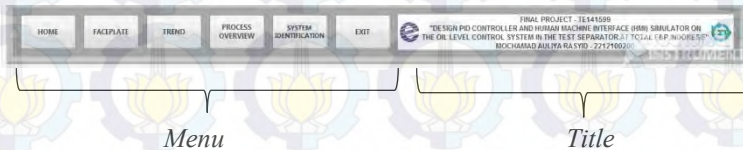
Bagian *body* merupakan bagian tengah dari HMI memiliki dimensi *pixels* dengan *height* sebesar 449 dan *width* 1272. Berisikan isi dari *window*. Pada Gambar 3.3 menunjukkan contoh bagian *body* pada *window home*.



Gambar 3.3 Tampilan Bagian *Body*

3.1.1.3 Bagian Footer

Bagian *header* merupakan bagian paling bawah dari HMI memiliki dimensi *pixels* dengan *height* sebesar 108 dan *width* 1272. Terdiri dari beberapa bagian seperti pada Gambar 3.4. Dan berisikan keterangan umum *window*.



Gambar 3.4 Tampilan Bagian *Footer*

Bagian *footer* terdiri dari *menu* dan *title*. Pada bagian menu berisikan menu-menu window yang akan muncul jika ditekan. Dan pada bagian title berisikan judul tugas akhir, nama, NRP penulis dan terdapat logo jurusan Teknik Elektro ITS serta bidang studi Teknik Sistem Pengaturan.

3.1.2 Window HMI

Pada HMI hasil perancangan terdapat beberapa *window* seperti *Home*, *Faceplate*, *Trend*, *Process Overview*, dan *System Identification*. *Window* tersebut akan muncul jika ditekan pada bagian *menu*.

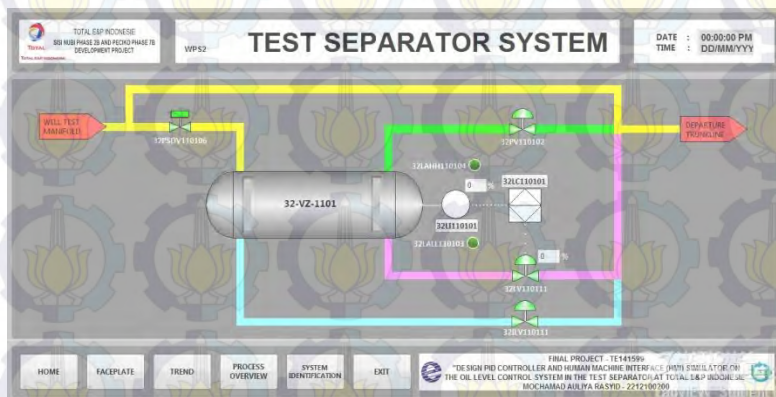
3.1.2.1 Window Home

Pada bagian *home* terdapat tampilan HMI seperti pada *real plant test separator 32-VZ-1101*. *Window* ini memiliki keterangan asal dan tujuan bahan yang diproses, bahan yang diproses dibedakan dengan warna sesuai dengan keterangan pada Tabel 3.1.

Tabel 3.1 Keterangan Warna HMI

No	Warna	Keterangan
1	Kuning	Fluida <i>Hydrocarbon</i>
2	Biru	Air
3	Ungu	Minyak
4	Hijau	Gas

Selain itu terdapat beberapa simbol sensor, kontroler dan *control valve*. Indikator isi *vessel* juga ditampilkan berikut besar nilai pada sensor dan *control valve*. Tampilan window home dapat dilihat pada Gambar 3.5.



Gambar 3.5 Tampilan Window Home

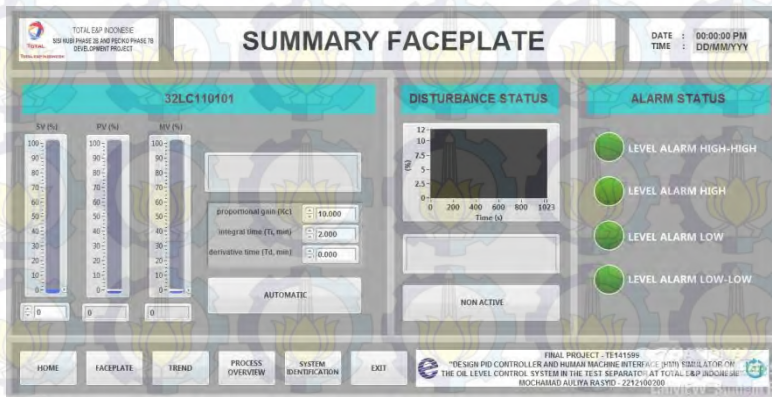
3.1.2.2 Window Faceplate

Pada bagian *faceplate* berguna untuk mengatur sistem dan tampilan *alarm*. Bagian yang *input* kontrol hanya pada bagian *oil level control* 32LC110101, pada bagian ini terdapat tampilan *bar* nilai *Set Point* (SP), *Process Value* (PV) dan *Manipulated Value* (MV). Selain itu operator dapat memilih mode kontrol apakah dengan mode *automatic* atau *manual*. Jika operator memilih model *automatic*, maka operator perlu mengatur SP dan parameter PID berupa K_p , T_i dan T_d . Nilai T_i dan T_d yang digunakan dalam satuan menit. Jika operator memilih mode *manual*, maka operator perlu menentukan besaran sinyal kontrol yang digunakan.

Pada *window* ini juga operator dapat menguji sistem dengan diberi gangguan dengan menekan tombol *active* pada bagian *disturbance status*. *Disturbance* yang muncul akan ditampilkan pada *trend disturbance*.

Pada bagian *alarm status* terdapat empat jenis alarm yaitu, *Level Alarm High-High* (LAHH), *Level Alarm High* (LAH), *Level Alarm Low* (LAL) dan *Level Alarm Low-Low* (LALL). Jika level berada lebih dari

80% maka LAHH akan aktif berwarna merah, jika level berada antara 70%-80% maka LAH akan aktif berwarna kuning dan berkedip, jika level berada anatar 30%-20% maka LAL akan aktif berwarna kuning dan berkedip, dan jika level berada dibawah 20% maka LALL akan aktif berwarna merah. Tampilan *window Faceplate* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Tampilan *Window Faceplate*

3.1.2.3 *Window Trend*

Pada bagian *trend* berguna untuk menampilkan grafik secara *real time* nilai SP, PV dan MV dari *oil level control* yang dapat diperbesar atau melihat data pada waktu sebelumnya. Grafik tersebut terdiri dari sumbu X berupa waktu dengan satuan detik dan sumbu Y berupa Level dengan satuan %, untuk membedakan variabel pada grafik, setiap variabel diberi warna yang berbeda agar dapat mudah membedakannya seperti yang dijelaskan pada Tabel 3.2.

Tabel 3.2 Keterangan Warna Grafik *Trend*

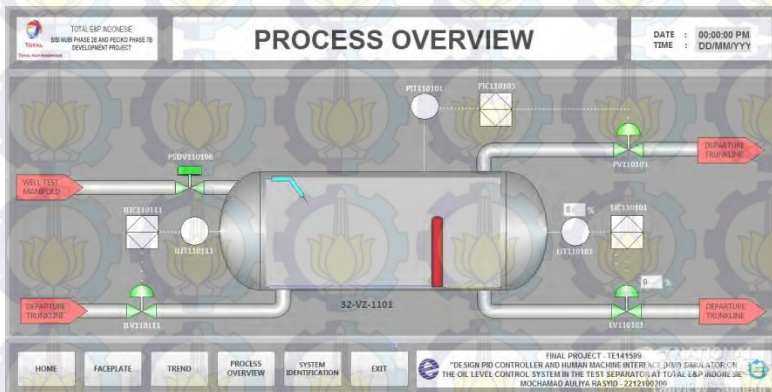
No	Warna	Keterangan
1	Putih	<i>Set Point</i>
2	Merah	<i>Process Value</i>
3	Hijau	<i>Manipulated Value</i>



Gambar 3.7 Tampilan *Window Trend*

3.1.2.4 Window Process Overview

Pada bagian *process overview* berguna untuk menampilkan animasi proses yang terjadi dalam *test separator*, beberapa bagian komponen dalam *test separator* juga dapat dilihat serta beberapa simbol komponen instrumentasi berupa sensor, kontroler dan *control valve*. Seperti yang ada pada Gambar 3.8.



Gambar 3.8 Tampilan *Window Process Overview*

3.1.2.5 Window System Identification

Pada bagian *system identification* berguna untuk menampilkan proses identifikasi sistem secara dinamis menggunakan metode Pendekatan Penyelesaian Persamaan Linier Simultan Parameter Total (P3LS-PT) dalam bentuk *Auto Regressive Moving Average* (ARMA) dengan memasukkan parameter input dari data real plant berupa time sampling dalam satuan detik, parameter kontroler PID, dan data *input* dan *output real plant*. Dari data yang didapat akan dihitung berapa jumlah data yang digunakan, parameter *output* berupa phi, theta dan *output* hasil identifikasi, serta model sistem dalam domain Laplace dan rekomendasi parameter kontroler. Perancangan hasil desain *window system identification* dapat dilihat seperti pada Gambar 3.9.

SYSTEM IDENTIFICATION

DATE : 00:00:00 PM
TIME : DD/MM/YYYY

PARAMETER INPUT

TIME SAMPLING
15

Kp
10

Ti (s)
120

TOTAL OF DATA
0

Data File Location
D:\data backup\Aula\Aula Materi Sistem Pengaturan\Tugas Akhir\Program Identifikasi\TSMO.txt

PARAMETER OUTPUT

PHI

THETA

OUTPUT

SYSTEM MODEL

CONTROLLER RECOMMENDATION

Kp
302.18

Ti
0

Td
0

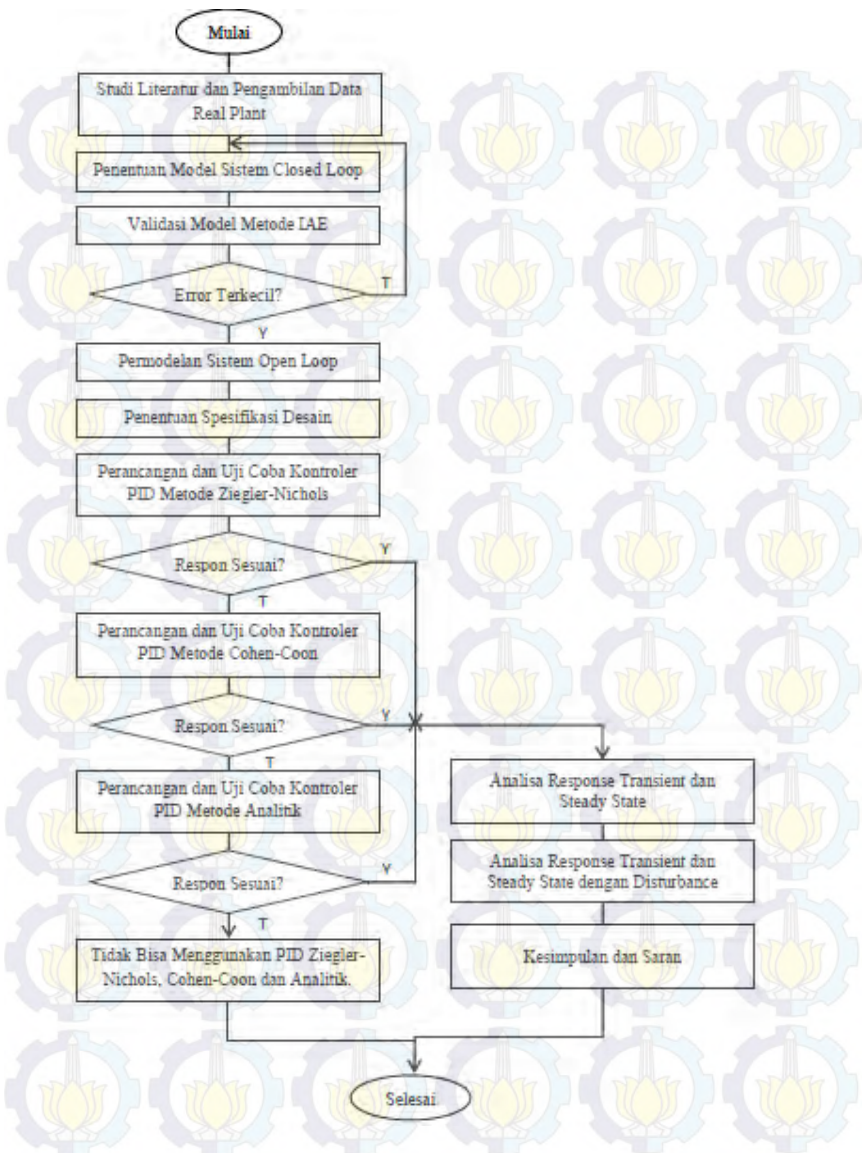
HOME FACEPLATE TREND PROCESS OVERVIEW SYSTEM IDENTIFICATION EXIT

FINAL PROJECT - 31411599
DESIGN PID CONTROLLER AND HUMAN MACHINE INTERFACE (HMI) FOR THE OIL LEVEL CONTROL SYSTEM IN THE TEST SEPARATOR AT TOTAL GAP INDONESIA
DOCHMAD AULYA RASYID - 2212100200

Gambar 3.9 Tampilan Window System Identification

3.2 Perancangan Kontroler PID Sistem Pengaturan Level

Penelitian ini dilakukan sesuai dengan diagram alir yang telah dibuat sesuai dengan *flowchart* Gambar 3.10. Diawali dengan studi literatur dan pengambilan data *real plant*, dilanjutkan dengan identifikasi sistem dimana terdiri dari penentuan model sistem *closed loop*. Model yang didapat akan dibandingkan eror terkecilnya dengan metode validasi IAE dan dipilih model dengan eror terkecil. Hasil model *closed loop* di cari model *open loop* dengan memasukkan parameter *controller* yang sudah ada sebelumnya. Selanjutnya dilanjutkan dengan tahap perancangan kontroler.



Gambar 3.10 *Flowchart* Metode Penelitian

Perancangan kontroler diawali dengan penentuan spesifikasi desain terlebih dahulu. Metode perancangan kontroler pertama menggunakan metode Ziegler-Nichols dalam bentuk P, PI dan PID. Parameter kontroler yang didapat kemudian diujikan menggunakan sinyal step. Jika tidak ada respons yang sesuai maka akan dilanjutkan dengan metode perancangan kontroler kedua metode Cohen-Coon dalam bentuk P, PI dan PID. Parameter kontroler yang didapat kemudian diujikan menggunakan sinyal step. Jika tidak ada respons yang sesuai maka akan dilanjutkan dengan metode perancangan kontroler ketiga dengan metode analitik dalam bentuk P, PI dan PID. Jika setelah pengujian tidak ada respons sistem yang memenuhi spesifikasi desain, maka dianggap sistem tidak dapat menggunakan kontroler PID metode Ziegler-Nichols, Cohen-Coon dan analitik.

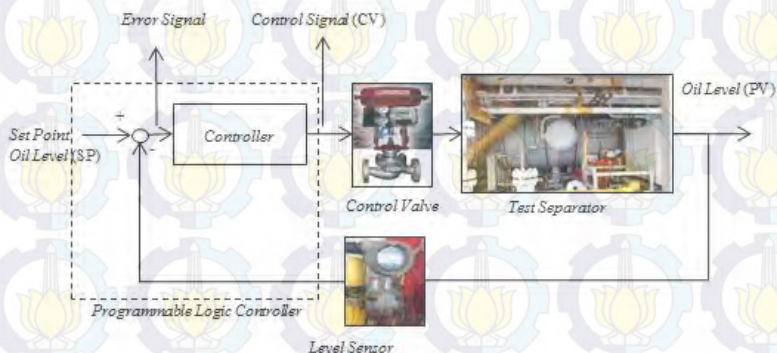
Jika terdapat respons yang sesuai dengan spesifikasi desain yang diinginkan, maka lebih lanjut akan dianalisis pada waktu *transient* dan waktu *steady state*. Serta pengujian dan analisis respons *transient* dan *steady state* dengan diberi *disturbance*. Hasil dari pengujian dan analisis dapat ditarik kesimpulan dan saran untuk parameter kontroler PID yang cocok bagi plant *Test Separator*.

3.3 Arsitektur Sistem Pengaturan

Arsitektur sistem pengaturan level minyak terdiri dari beberapa komponen seperti pada Gambar 3.11 yaitu, *Controller* berupa *Programmable Logic Controller* (PLC) yang juga merupakan sebuah *Remote Terminal Unit* (RTU) pada sistem *Distributed Control System* (DCS) dengan memiliki *Human Machine Interface* (HMI) sebagai interaksi untuk memasukan nilai *set point* (SV/*Set Value*), menampilkan besar sinyal kontrol (CV/*Control Value*) dan ketinggian level (PV/*Process Value*) serta memasukan parameter Kontroler PID, *Control Valve* sebagai aktuator yang menerima sinyal kontrol yang dikirim oleh *controller*, Sensor level yang digunakan untuk mengetahui ketinggian level minyak didalam *vessel* serta Plant *Test Separator*.

Set point level Minyak diberikan pada PLC di HMI dalam range 0-100% kemudian nilai tersebut akan dibandingkan dengan nilai yang dibaca oleh sensor level pada range 0-100% oleh komparator sehingga menghasilkan sinyal *error*. Sinyal *error* akan masuk ke kontroler dan akan diolah dengan algoritma kontroler sehingga menghasilkan sinyal kontrol. Sinyal kontrol akan dikirimkan ke *control valve* untuk membuka atau menutup sesuai dengan perintah kontroler, sehingga level

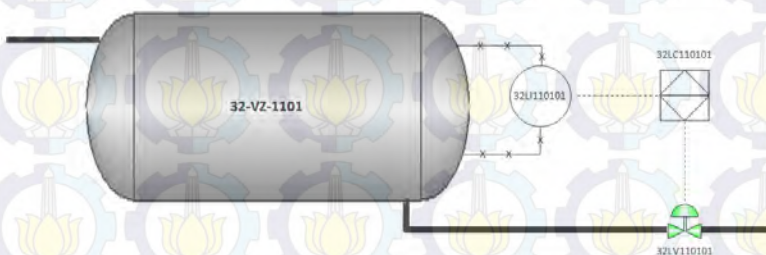
minyak pada *test separator* akan naik atau turun, kejadian ini akan terus berlangsung hingga nilai *process value* sesuai dengan nilai *set point*.



Gambar 3.11 Arsitektur Sistem Pengaturan *Oil Level*

Level minyak diatur sesuai dengan *set point* karena jika terlalu tinggi, minyak dapat masuk ke saluran gas dan jika terlalu rendah gas dapat masuk ke saluran minyak. Hal ini dapat mengganggu proses *data logging* pengukuran setiap fasa dari sumur. pengaturan level minyak pada *test separator* dengan cara mengatur bukaan *valve* pada *outlet* saluran minyak dimana besar bukaan *valve* tergantung dari sinyal *control* yang dikirim oleh kontroler.

3.3.1 P&ID Sistem Pengaturan Level



Gambar 3.12 P&ID Sistem Pengaturan Level *Test Separator*

Untuk menunjang berjalannya sistem pengaturan *level* minyak, dibutuhkan komponen instrumentasi berupa sensor, *actuator* dan kontroler seperti pada Gambar 3.12. Dari P&ID dapat dilihat bahwa level dari *test separator* 32-VZ-1101 diukur oleh sensor berupa *level transmitter* 32LI110101, hasilnya berupa sinyal listrik dikirim ke *level controller* 32LC110101. Keluaran dari kontroler berupa sinyal *control* dikirimkan ke *level control valve* 32LV11010 dan akan mengatur besar bukaan *valve* pada *oil outlet* daftar instrument yang digunakan dapat dilihat pada Tabel 3.3.

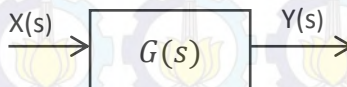
Tabel 3.3 *Instrumentation Code List*

No	Kode	Keterangan
1	32-VZ-1101	<i>Test Separator</i>
2	32LI110101	<i>Level Indicator</i>
3	32LV110101	<i>Level Control Valve</i>
4	32LC110101	<i>Level Controller</i>

3.4 Identifikasi Sistem

Identifikasi sistem merupakan proses yang penting dalam siklus perancangan sistem pengaturan. Hal tersebut diperlukan untuk mendapatkan model matematika dari sistem. Pada bagian ini akan dibahas bagaimana mendapatkan model *closed loop* hingga *open loop* serta transformasi model dari domain waktu diskrit ke domain Z ke domain Laplace.

3.4.1 Identifikasi Sistem *Closed Loop*



Gambar 3.13 Identifikasi Sistem

Proses identifikasi sistem *closed loop* menggunakan cara identifikasi dinamis bentuk ARMA metode P3LS-PT menggunakan data dari *Distributed Control System* (DCS) yang dilakukan secara *closed*

loop dengan sudah terdapat parameter PID sebelumnya. Sehingga akan mendapatkan fungsi alih dari sistem seperti pada Gambar 3.13.

Model yang didapat bentuk ARMA dalam domain waktu diskrit ditransformasi ke domain Z, setelah itu model yang didapatkan ditransformasikan kembali ke domain Laplace sehingga didapatkan beberapa model sebagai berikut

Identifikasi model pertama didapatkan dalam domain waktu diskrit seperti pada Persamaan 3.1

$$y(k) = 0.976523y(k-1) + 0.0259457x(k-1) \quad (3.1)$$

Persamaan model pertama hasil transformasi ke domain Z dapat ditulis dalam bentuk seperti pada Persamaan 3.2.

$$\frac{Y(z)}{X(z)} = \frac{0.0259457}{z - 0.976523} \quad (3.2)$$

Persamaan model pertama hasil transformasi ke domain Laplace dapat ditulis dalam bentuk seperti pada Persamaan 3.3.

$$\frac{Y(s)}{X(s)} = \frac{1.10514}{638.915s + 1} \quad (3.3)$$

Identifikasi model kedua didapatkan dalam domain waktu diskrit seperti pada Persamaan 3.4.

$$y(k) = 1.03091y(k-1) - 0.0537135y(k-2) + 0.025066x(k-1) \quad (3.4)$$

Persamaan model kedua hasil transformasi ke domain Z dapat ditulis dalam bentuk seperti pada Persamaan 3.5.

$$\frac{Y(z)}{X(z)} = \frac{0.025066}{z^2 - 1.03091z + 0.0537135} \quad (3.5)$$

Persamaan model kedua hasil transformasi ke domain Laplace dapat ditulis dalam bentuk seperti pada Persamaan 3.6

$$\frac{Y(s)}{X(s)} = \frac{0.025066}{255s^2 + 14.5364s + 0.022808} \quad (3.6)$$

Identifikasi model ketiga didapatkan dalam domain waktu diskrit seperti pada Persamaan 3.7.

$$y(k) = 1.03091y(k-1) - 0.0360545y(k-2) + 0.0126934x(k-1) + 0.0126934x(k-2) \quad (3.7)$$

Persamaan model ketiga hasil transformasi ke domain Z dapat ditulis dalam bentuk seperti pada Persamaan 3.8.

$$\frac{Y(z)}{X(z)} = \frac{0.0126934z + 0.0126934}{z^2 - 1.03659z + 0.0360545} \quad (3.8)$$

Persamaan model ketiga hasil transformasi ke domain Laplace dapat ditulis dalam bentuk seperti pada Persamaan 3.9.

$$\frac{Y(s)}{X(s)} = \frac{0.190401s + 0.0253868}{255s^2 + 14.451s - 0.000532449} \quad (3.9)$$

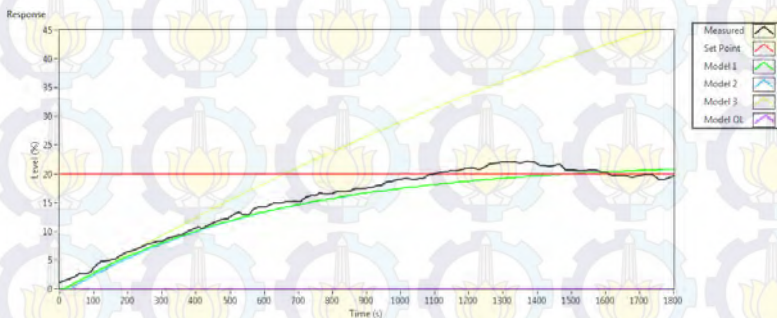
3.4.2 Validasi Model Sistem *Closed Loop*

Pada hasil pengujian pengaturan level *oil*, dengan cara memberi masukan perubahan *set point* sebesar 20% sesuai dengan yang dilakukan pada saat pengambilan data *real*. Didapatkan model estimasi dengan membandingkan *input* dan *output* dari sistem, menggunakan model yang didapat dengan data yang didapat dari DCS dengan perlakuan yang sama. Orde sistem diubah secara bertahap dan dipilih model yang memiliki *error* terkecil melalui metode IAE. Dari hasil simulasi didapatkan beberapa model yang dibandingkan dengan data asli seperti pada Tabel 3.4.

Tabel 3.4 Validasi Model

No	Order		Estimated Model	Error
	Num	Den		
1	0	1	$\frac{Y(s)}{X(s)} = \frac{1.10514}{638.915s + 1}$	7.163
2	0	2	$\frac{Y(s)}{X(s)} = \frac{0.025066}{255s^2 + 14.5364s + 0.02280}$	7.68582
3	1	2	$\frac{Y(s)}{X(s)} = \frac{0.190401s + 0.0253868}{255s^2 + 14.451s - 0.0005324}$	67.9838

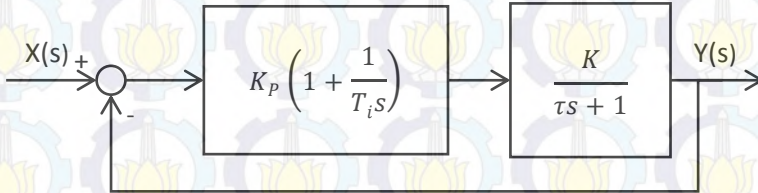
Hasil dari simulasi tiap model estimasi dan dibandingkan dengan data *real plant* dapat dilihat seperti pada Gambar 3.14.



Gambar 3.14 Respons Hasil Model Estimasi

Hasil dari validasi model didapatkan model dengan *error* terkecil adalah dengan nilai *numerator* 0 dan *denominator* 1. Sehingga model tersebut yang dipilih untuk tahap selanjutnya. Model matematis yang

dipilih dan perbandingan model lain dapat dilihat seperti pada Persamaan 3.10.



Gambar 3.15 Blok Diagram Sistem *Closed Loop*

$$\frac{Y(s)}{X(s)} = \frac{1.10514}{638.915s + 1} \quad (3.10)$$

3.4.3 Identifikasi Sistem *Open Loop*

Dengan memiliki model matematis *closed loop*, dapat menentukan model *open loop* dengan mengetahui parameter PID yang didapatkan pada metode *trial and error* seperti blok diagram pada Gambar 3.15.

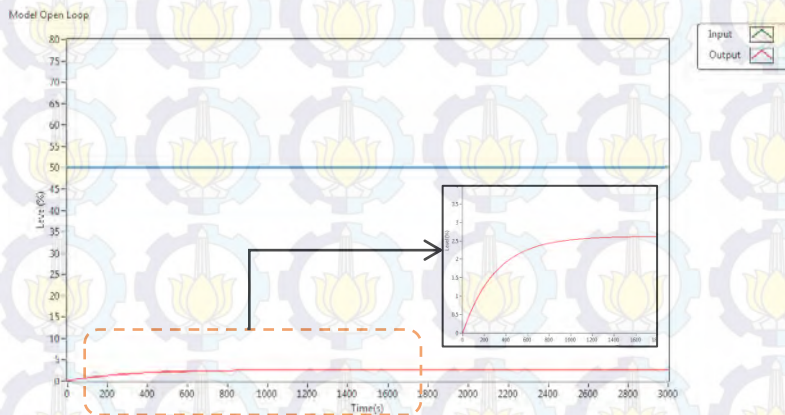
Dengan diketahui nilai K_p dan T_i seperti pada Persamaan 3.11 dapat diketahui nilai *open loop* dari sistem.

$$\begin{aligned} K_p &= 10 \\ T_i &= 2 \text{ Menit} = 120 \text{ Detik} \end{aligned} \quad (3.11)$$

Setelah melakukan perhitungan secara analitik dapat ditentukan nilai model *open loop* yang didekatkan ke persamaan orde satu seperti pada Persamaan 3.12.

$$\frac{Y(s)}{X(s)} = \frac{0.0522685}{302.18s + 1} \quad (3.12)$$

Model yang telah didapatkan akan menjadi acuan sebagai penentuan parameter kontroler. Respons step dari model *open loop* dapat dilihat seperti Gambar 3.16.



Gambar 3.16 Respons Step *Open Loop*

3.5 Spesifikasi Desain Sistem

Sistem yang dirancang diharapkan memiliki spesifikasi desain sebagai berikut

1. Memiliki nilai *error steady state* sebesar 0% atau *zero offset* karena jika *level* minyak terlalu tinggi atau terlalu rendah akan mengganggu proses seperti masuknya minyak ke *outlet* gas dan mengaktifkan *alarm high-high* atau *low-low* yang dapat membuat *system shutdown*.
2. Tidak memiliki *overshoot* karena jika level minyak terlalu tinggi atau terlalu rendah akan mengganggu proses seperti masuknya minyak ke *outlet* gas dan mengaktifkan *alarm high-high* atau *low-low* yang dapat membuat *system shutdown*.
3. Memiliki *time constant* lebih cepat dari sistem *real* untuk mempercepat proses.

3.6 Perancangan Kontroler PID

Perancangan Kontroler PID terdiri dari beberapa metode yaitu metode Ziegler-Nichols, Cohen-Coon dan secara analitik.

3.6.1 Kontroler PID Metode Ziegler-Nichols

Dari data yang didapat, dapat ditentukan nilai $L=7.73$ dan $T=329$. Kemudian data tersebut dimasukan kedalam tabel dan didapatkanlah parameter PID seperti pada Tabel 3.5.

Tabel 3.5 Hasil Parameter Kontroler PID Ziegler-Nichols

Type	Kp	Ti	Td
P	42.56	-	-
PI	38.31	25.71	-
PID	51.07	15.46	3.865

3.6.2 Kontroler PID Metode Cohen-Coon

Dari hasil penalaan secara grafis, dan dimasukan ke tabel penalaan kontroler metode Cohen-Coon akan didapatkan nilai parameter PID seperti pada Tabel 3.6.

Tabel 3.6 Hasil Parameter Kontroler PID Cohen-Coon

Type	Kp	Ti	Td
P	3008.66	-	-
PI	2703.64	6.288	-
PID	4007.82	4.69366	0.6944

3.6.3 Kontroler PID Metode Analitik

Dari model yang didapat, ditentukan kontroler menggunakan tipe PI saja karena kontroler jenis ini sudah bisa mengakomodir spesifikasi desain sesuai dengan keinginan. Parameter ini didapat dengan perhitungan secara matematis. Ditentukan spesifikasi desain *zero offset* dengan *settling time* yang bisa diubah-ubah sehingga menghasilkan nilai sebagai T_i sebesar 302.18 s, seperti pada Persamaan 3.13 dan nilai K_p yang berubah ubah sesuai dengan *time constant* sesuai dengan Tabel 3.7.

$$\begin{aligned} T_i &= \tau \\ K_p &= \frac{5811.15}{\tau^*} \end{aligned} \quad (3.13)$$

Tabel 3.7 Hasil Parameter Kontroler PI Analitik

<i>Time Constant</i>	Kp
5τ	3.846151
2τ	9.615378
τ	19.23076
0.8τ	24.03845
0.6τ	32.05126

BAB IV

PENGUJIAN DAN ANALISIS

Pada Bab ini membahas tentang simulasi dan analisis sistem yang telah dirancang. Pengujian dan analisis meliputi uji coba HMI hasil perancangan, simulasi sistem *open loop* dan *closed loop* hasil identifikasi sistem, simulasi sistem menggunakan kontroler PID metode Ziegler-Nichols, Cohen-Coon dan Analitik.

4.1 Pengujian Human Machine Interface (HMI)

Pengujian HMI bertujuan untuk mengetahui apakah HMI yang dirancang sudah sesuai dengan yang diinginkan, pengujian HMI berupa pengujian fitur yang ada didalamnya seperti pengujian *call window menu*, pengujian menu kontrol, pengujian *disturbance*, pengujian *alarm status*, pengujian identifikasi sistem dan pengujian indikator-indikator pada HMI.

4.1.1 Uji Call Window Menu

Pengujian *call window menu* bertujuan apakah *menu* yang sudah dibuat jika ditekan akan menampilkan *window* yang sesuai.

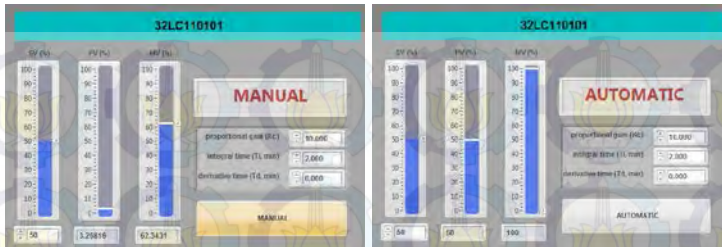
Tabel 4.1 Pengujian *Call Window Menu*

No	Tombol yang ditekan	Window yang ditampilkan
1	<i>Home</i>	<i>Home</i>
2	<i>Faceplate</i>	<i>Faceplate</i>
3	<i>Trend</i>	<i>Trend</i>
4	<i>Process Overview</i>	<i>Process Overview</i>
5	<i>System Identification</i>	<i>System Identification</i>
6	<i>Exit</i>	Program Berhenti

Dari Tabel 4.1 hasil pengujian yang dilakukan, *program menu call window* sudah berjalan sesuai dengan keinginan. *Window* yang dipanggil akan keluar sesuai dengan tombol yang ditekan.

4.1.2 Uji Control Menu

Pengujian *Control Menu*, bertujuan apakah *control menu* yang dibuat sudah dapat berjalan sesuai *mode* yang diinginkan.

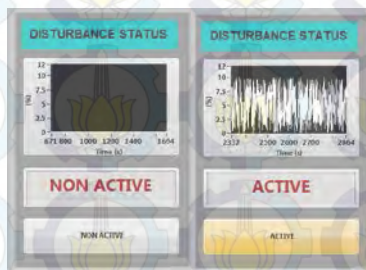


Gambar 4.1 Pengujian *Control Mode Manual* dan *Automatic*

Dari hasil pengujian Gambar 4.1 dapat dilihat bahwa jika tombol manual ditekan, akan menampilkan status mode kontrol *manual* dan nilai MV dapat diubah sesuai keinginan. dan jika tombol kontrol automatic ditekan, akan menampilkan status mode *automatic* dan memasukan nilai SP serta parameter PID akan mempengaruhi nilai PV.

4.1.3 Uji *Disturbance Menu*

Pengujian *Disturbance Menu*, bertujuan apakah *Disturbance menu* yang dibuat akan menghasilkan *disturbance* bagi sistem yang dibuat.



Gambar 4.2 Pengujian *Disturbance Menu Non Active* dan *Active*

Dari hasil pengujian Gambar 4.2 dapat dilihat bahwa jika tombol *Non Active* ditekan, akan menampilkan status mode *disturbance Non Active* dan pada grafik tidak terdapat grafik *disturbance*. Dan jika tombol *Active* ditekan, akan menampilkan status mode *Active* dan pada

grafik akan terlihat grafik *disturbance*, aktifnya *disturbance* juga mempengaruhi nilai PV.

4.1.4 Uji Alarm Status

Pengujian *Alarm Status* bertujuan untuk mengetahui apakah alarm yang dibuat akan aktif jika terdapat anomali pada level proses. Hasil pengujian Alarm Status dilakukan dengan mengubah *set point* berada dalam *range alarm*.



Gambar 4.3 Pengujian Alarm Status

Dari hasil pengujian Gambar 4.3 dapat dilihat bahwa setiap alarm sudah dapat bekerja jika terjadi anomali pada level proses. Aktifnya LAHH akan menyalakan indikator berwarna merah dimana pada *real system* akan mengakibatkan *shutdown system*. Aktifnya alarm LAHH dapat mengakibatkan minyak masuk ke saluran gas. Aktifnya LAH akan menyalakan indikator berwarna kuning yang berkedip, dimana pada *real system* indikator ini berfungsi sebagai peringatan awal kepada operator sebelum proses menuju LAHH sehingga operator dapat melakukan aksi agar proses tidak sampai menyalakan LAHH. Aktifnya LAL akan menyalakan indikator berwarna kuning yang berkedip, dimana pada *real system* indikator ini berfungsi sebagai peringatan awal kepada operator sebelum proses menuju LALL sehingga operator dapat melakukan aksi agar proses tidak sampai menyalakan LALL. Aktifnya LALL akan menyalakan indikator berwarna merah dimana pada *real system* akan mengakibatkan *shutdown system*. Aktifnya alarm LAHH dapat mengakibatkan gas masuk ke saluran minyak. Jika terdapat fasa yang keluar pada *outlet* yang tidak sesuai pada *test separator* maka akan mengakibatkan proses pemisahan serta pengujian sumur minyak dan gas tidak berlangsung dengan baik.

4.1.5 Uji System Identification Menu

Pengujian *System Identification Menu*, bertujuan apakah program identifikasi yang dibuat sudah berjalan dengan baik jika diberi parameter *input* akan menghasilkan parameter *output*.



Gambar 4.4 Pengujian System Identification

Dari hasil pengujian Gambar 4.4 dapat dilihat bahwa dengan memasukan data dari *real plant* berupa nilai *SP*, *PV*, *Time Sampling* dan parameter kontroler maka akan menjalankan program identifikasi dan menghasilkan model sistem dalam bentuk Laplace dan rekomendasi parameter kontroler yang digunakan.

4.1.6 Uji Indikator HMI

Pengujian indikator HMI bertujuan apakah indikator-indikator yang ada pada HMI memiliki nilai yang sama pada variabel yang sama, daftar indikator dan hasil pengujian dapat dilihat seperti pada Tabel 4.2.

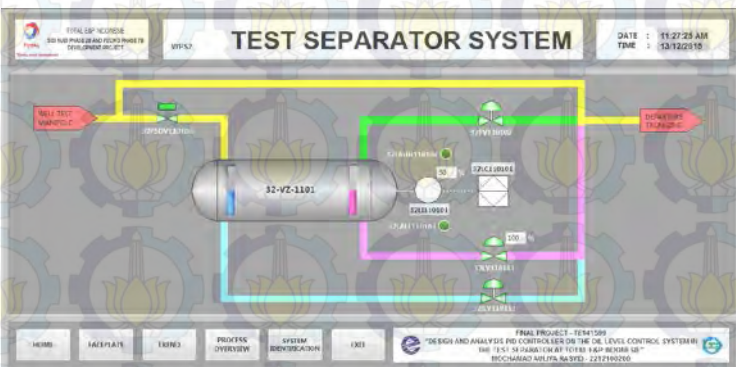
Tabel 4.2 Pengujian Indikator HMI

No	Type	Window	Nilai
1	Waktu	Home	Date : 11:27:25 AM Time : 13/12/2015
		Faceplate	Date : 11:27:25 AM Time : 13/12/2015
		Trend	Date : 11:27:25 AM Time : 13/12/2015
		Process Overview	Date : 11:27:25 AM Time : 13/12/2015

Tabel 4.2 Pengujian Indikator HMI (lanjutan)

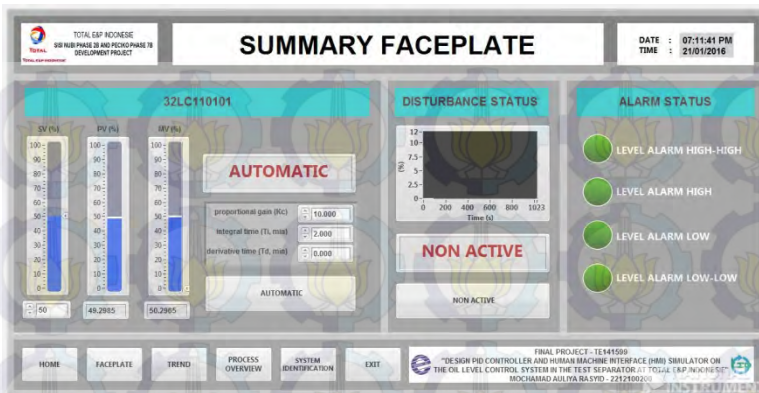
No	Type	Window	Nilai
		<i>System Identification</i>	<i>Date : 11:27:25 AM</i> <i>Time : 13/12/2015</i>
2	SP	<i>Faceplate</i>	50%
		<i>Trend</i>	50%
3	PV	<i>Home</i>	50%
		<i>Faceplate</i>	50%
		<i>Trend</i>	50%
		<i>Process Overview</i>	50%
4	MV	<i>Home</i>	100%
		<i>Faceplate</i>	100%
		<i>Trend</i>	100%
5	LAHH	<i>Home</i>	Normal
		<i>Faceplate</i>	Normal
6	LALL	<i>Home</i>	Normal
		<i>Faceplate</i>	Normal

Hasil pengujian indikator pada HMI setiap *window* dapat dilihat seperti pada Gambar 4.5, 4.6, 4.7, 4.8, 4.9.



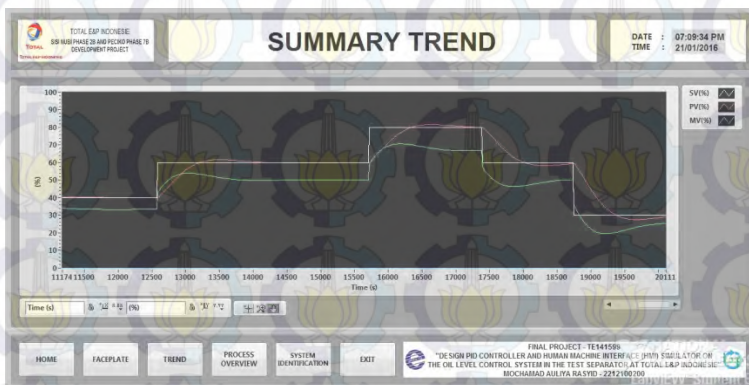
Gambar 4.5 Pengujian Indikator Window Home

Window home pada Gambar 4.5 berjalan sesuai dengan program yang telah dirancang, semua indikator terhubung dengan indikator lain seperti indikator PV, MV, LAHH dan LALL.



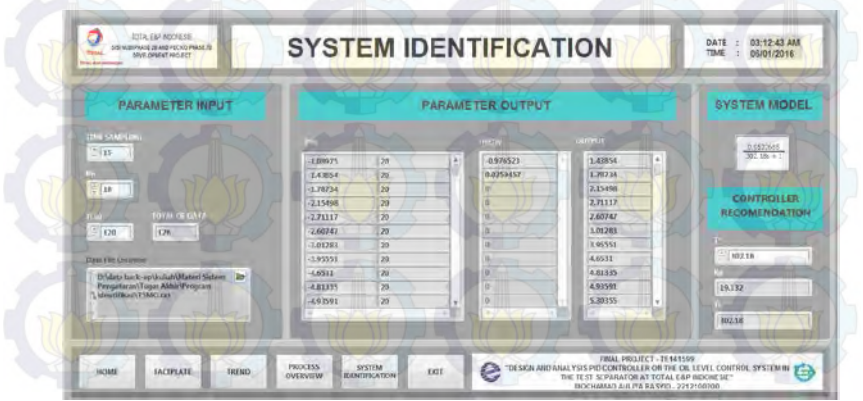
Gambar 4.6 Pengujian Indikator *Window Faceplate*

Window faceplate pada Gambar 4.6 berjalan sesuai dengan program yang telah dirancang, semua indikator terhubung dengan indikator lain seperti indikator SV, PV, MV, LAHH, LAH, LAL, LALL serta menu lain berfungsi sesuai dengan program yang dirancang.



Gambar 4.7 Pengujian Indikator *Window Trend*

Window trend pada Gambar 4.7 berjalan sesuai dengan program *real time trend* yang telah dirancang, semua indikator terhubung dengan

[illegible]

Gambar 4.9 Pengujian Indikator *Window System Identification*

Window System Identification pada Gambar 4.9 berjalan sesuai dengan program identifikasi yang telah dirancang, model yang menjadi *output* sistem akan menjadi model matematis yang dijalankan pada seluruh program HMI.

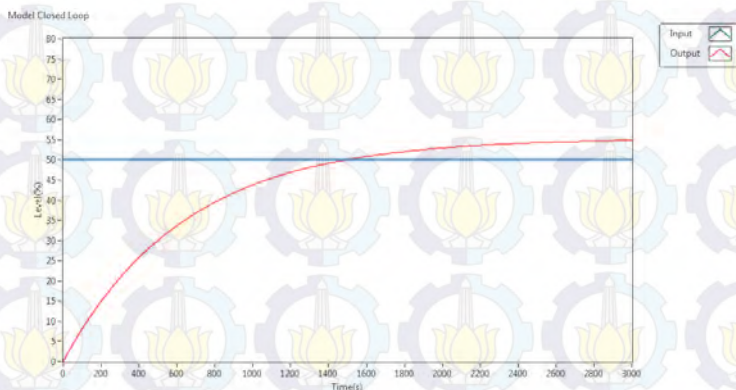
Setiap *window* dari HMI yang dirancang juga menunjukkan waktu yang sama pada indikator waktu.

4.2 Pengujian Model

Pengujian model bertujuan untuk mendapatkan model yang sesuai dengan *real plant* serta mengetahui karakteristik baik dari kondisi *transient* maupun *steady state* dari model tersebut. Pengujian tersebut meliputi pengujian *closed loop* dan *open loop*.

4.2.1 Uji Closed Loop

Pengujian model *closed loop* bertujuan untuk mengetahui karakteristik respons sistem untuk analisis pada daerah *transient* maupun daerah *steady state* dari hasil permodelan *real* sistem. Dari respons step dengan memberikan sinyal masukan sebesar 50% dapat dilihat respons sistem seperti pada Gambar 4.10.



Gambar 4.10 Respons Sistem *Closed Loop*

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons pada daerah *transient* maupun *steady state* seperti pada Tabel 4.3.

Tabel 4.3 Karakteristik Sistem *Closed Loop*

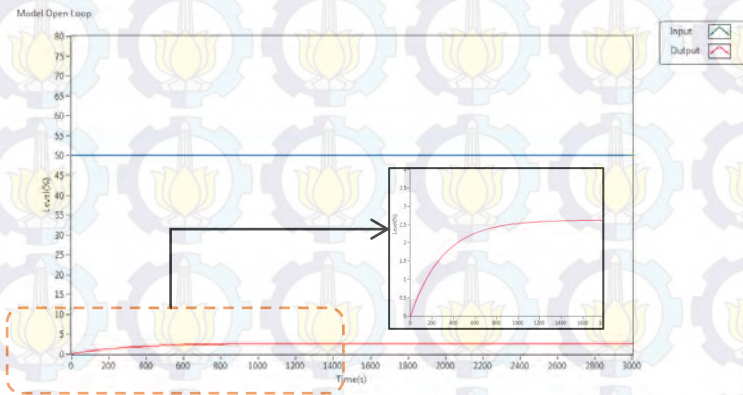
No	Nama	Nilai
1	Y_{ss}	55.25
2	X_{ss}	50.00
3	E_{ss}	5.25
4	$E_{ss}\%$	10.50
5	τ	638.73
6	K	1.105
7	$t_r(5\% - 95\%)$	1880.702
8	$t_r(10\% - 90\%)$	1403.433
9	t_d	442.7339
10	$t_s(5\%)$	1916.19
11	$t_s(2\%)$	2554.92
12	$t_s(0.5\%)$	3193.65

Dari respons yang didapat, dapat dilihat bahwa respons memiliki *error* pada saat kondisi *steady state* sebesar 10.5%, dalam kondisi *real* hal ini dapat mengganggu proses karena dapat mengakibatkan fasa minyak masuk ke *outlet* gas atau dapat mengaktifkan *level alarm high-high*. Akibat dari aktifnya *level alarm high-high* dapat mengakibatkan *shutdown system*. *Shutdown system* merupakan hal yang sangat dihindari di lapangan karena mengganggu proses lain yang sedang berjalan. Selain itu sistem memiliki *time constant* yang besar yaitu sebesar 638.73 s, hal tersebut tidak terlalu berpengaruh dalam jalannya proses, Namun *time constant* yang besar akan mengakibatkan kurang efektifnya waktu, sehingga untuk meningkatkan tingkat efektifitas waktu *time constant* perlu dipercepat.

4.2.2 Uji *Open Loop*

Pengujian *open loop* bertujuan untuk melihat karakteristik respons tanpa menggunakan kontroler, sehingga dapat ditentukan kontroler yang dibutuhkan sesuai dengan spesifikasi desain yang telah ditentukan sebelumnya. Respons *open loop* diperlukan karena perancangan kontroler metode Ziegler-Nichols dan Cohen-Coon merupakan metode *based on graphic*, sehingga harus mengetahui terlebih dahulu bagaimana respons *open loop* sistem. Pengujian sistem *open loop* dilakukan dengan cara memberi sinyal masukan berupa step

sebesar 50%, dan respons *open loop* hasil simulasi dapat dilihat seperti pada gambar 4.11.



Gambar 4.11 Respons Sistem *Open Loop*

Dari respons yang didapat dapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.4.

Tabel 4.4 Karakteristik Sistem *Open Loop*

No	Nama	Nilai
1	Y_{SS}	2.50
2	X_{SS}	50.00
3	E_{SS}	47.50
4	$E_{SS}\%$	94.78
5	τ	302.73
6	K	0.05
7	$t_r(5\% - 95\%)$	891.3700122
8	$t_r(10\% - 90\%)$	665.1657963
9	t_d	209.836446
10	$t_s(5\%)$	908.19
11	$t_s(2\%)$	1210.92
12	$t_s(0.5\%)$	1513.65

Dari respons yang didapat, dapat dilihat bahwa respons *open loop* memiliki *error steady state* yang besar yaitu 94.78%, hal ini dapat mengakibatkan level didalam *vessel* kosong. Hal tersebut dapat terjadi karena minyak yang keluar lebih banyak dibandingkan minyak yang masuk. Sehingga mengakibatkan *vessel* kosong. Hal tersebut dalam kondisi *real* sangat berbahaya. Karena gas dapat masuk ke *outlet* minyak dan juga level yang sangat rendah dapat mengakibatkan *alarm level low-low* aktif sehingga mengakibatkan *system shutdown*.

Dari respons *open loop* yang didapat, dapat disimpulkan bahwa sistem membutuhkan kontroler untuk menjaga level minyak dalam *vessel* sesuai dengan *set point*.

4.3 Pengujian Kontroler PID

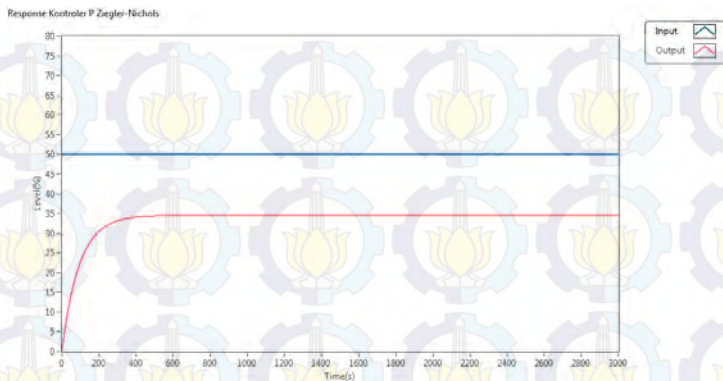
Pengujian kontroler PID yang telah didapatkan pada bab sebelumnya bertujuan untuk mendapatkan parameter kontroler PID yang cocok sesuai dengan spesifikasi desain yang telah ditentukan. Dari beberapa metode penentuan kontroler PID dapat dianalisis kontroler mana yang memiliki respons yang sesuai dengan spesifikasi desain yang telah ditentukan. Metode yang diuji yaitu metode Ziegler-Nichols, Cohen-Coon dan Analitik. Sesuai dengan diagram alur metode penelitian yang telah dirancang, kontroler yang digunakan pertama adalah tipe P terlebih dahulu, jika respons belum sesuai dengan spesifikasi desain maka dilanjutkan kontroler tipe PI dan menggunakan kontroler tipe PID jika kontroler jenis PI belum dapat memenuhi spesifikasi desain yang telah ditentukan dan proses ini dilakukan sampai mendapatkan respons yang sesuai. Jika tidak terdapat respons yang sesuai maka kontroler yang digunakan tidak dapat memenuhi spesifikasi desain yang ditentukan.

4.3.1 Kontroler Metode Ziegler-Nichols

Dengan memasukkan nilai parameter kontroler PID sesuai dengan tabel dari metode Ziegler-Nichols, dapat ditentukan karakteristik respons menggunakan kontroler tipe P, PI dan PID.

4.3.1.1 Kontroler Tipe P

Dengan memasukkan parameter kontroler tipe P berupa K_p sebesar 42.46 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal step sebesar 50% sesuai dengan gambar yang ada pada Gambar 4.3.



Gambar 4.12 Respons Kontroler Tipe P Metode Ziegler-Nichols

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.5.

Tabel 4.5 Karakteristik Kontroler P Metode Ziegler-Nichols

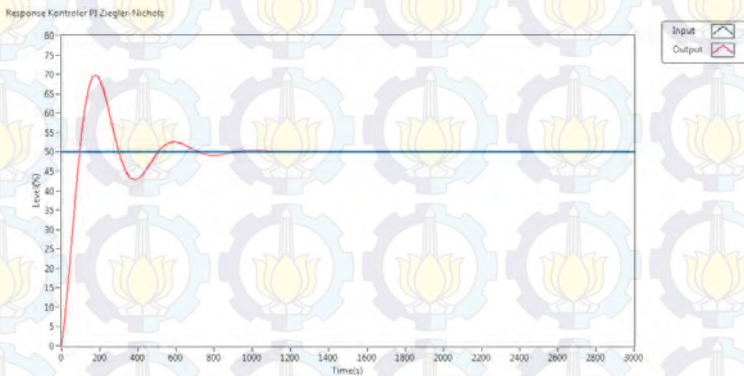
No	Nama	Nilai
1	Y_{SS}	34.50
2	X_{SS}	50.00
3	E_{SS}	15.50
4	$E_{SS} \%$	31.00
5	τ	302.73
6	K	0.69
7	$t_r(5\% - 95\%)$	891.3700122
8	$t_r(10\% - 90\%)$	665.1657963
9	t_d	209.836446
10	$t_s(5\%)$	908.19
11	$t_s(2\%)$	1210.92
12	$t_s(0.5\%)$	1513.65

Dari respons yang didapat, respons sistem masih memiliki *error steady state* sebesar 31%, sehingga dengan menggunakan kontroler tipe

P metode Ziegler-Nichols masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.1.2 Kontroler Tipe PI

Dengan memasukkan parameter kontroler P berupa K_p sebesar 38.31 serta T_i sebesar 25.71 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal step sebesar 50% sesuai dengan gambar yang ada pada Gambar 4.13



Gambar 4.13 Respons Kontroler Tipe PI Metode Ziegler-Nichols

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde dua dan dapat diketahui karakteristik respons seperti pada Tabel 4.6.

Tabel 4.6 Karakteristik Kontroler PI Metode Ziegler-Nichols

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	K	1.00
6	M_p	69.75
7	$M_p\%$	39.50

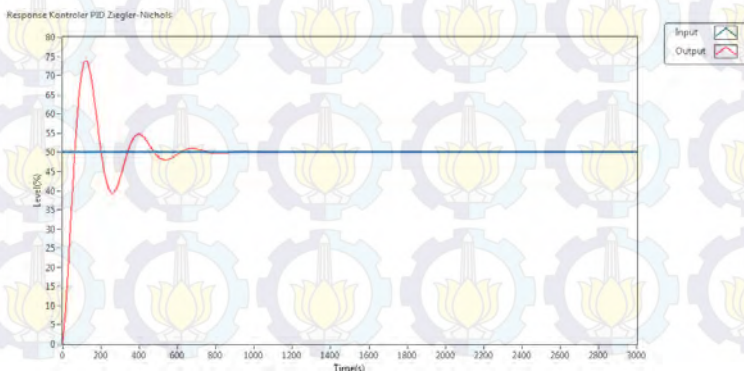
Tabel 4.6 Karakteristik Kontroler PI Metode Ziegler-Nichols (lanjutan)

No	Nama	Nilai
8	t_p	176.73
9	t_d	50.73
10	ζ	0.283666948
11	w_n	0.018528305
12	τ	190.2635379
13	$t_s(5\%)$	570.7906137
14	$t_s(2\%)$	761.0541516
15	$t_s(0.5\%)$	951.3176895

Dari respons yang didapat, dapat dilihat bahwa respons sudah tidak memiliki *error steady state*, namun memiliki *overshoot* sebesar 39.50% yang dapat mengakibatkan aktifnya *level alarm high-high* pada awal sistem dimulai. Sehingga dengan menggunakan kontroler tipe PI metode Ziegler-Nichols masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan. Sehingga dilanjutkan sesuai dengan menggunakan kontroler PID.

4.3.1.3 Kontroler Tipe PID

Dengan memasukan parameter kontroler PID berupa Kp sebesar 51.07, Ti sebesar 15.46 dan Td sebesar 3.865 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal step sebesar 50% sesuai Gambar 4.14.



Gambar 4.14 Respons Kontroler Tipe PID Metode Ziegler-Nichols

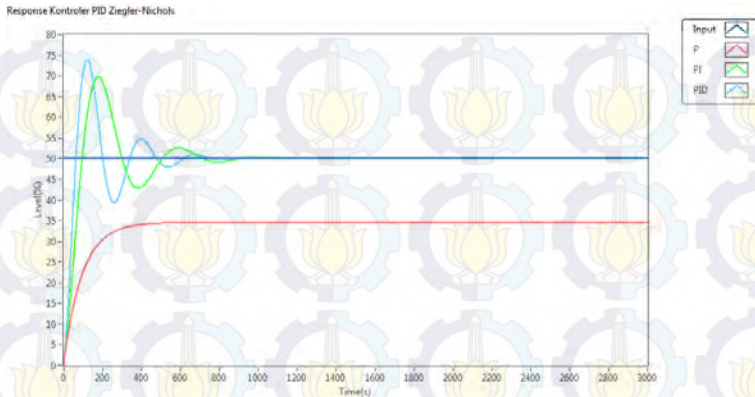
Dari respons yang didapat, respons dapat didekatkan ke bentuk orde dua dan dapat diketahui karakteristik respons pada daerah *transient* dan *steady state* seperti pada Tabel 4.7.

Tabel 4.7 Karakteristik Kontroler PID Metode Ziegler-Nichols

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	K	1.00
6	M_p	73.825
7	$M_p\%$	47.65
8	t_p	120.624
9	t_d	35.624
10	ζ	0.229762934
11	w_n	0.026746873
12	τ	162.7222783
13	$t_s(5\%)$	488.166835
14	$t_s(2\%)$	650.8891134
15	$t_s(0.5\%)$	813.6113917

Dari respons yang didapat, dapat dilihat bahwa respons sudah tidak memiliki *error steady state*, namun memiliki *overshoot* sebesar 47.65% yang dapat mengakibatkan aktifnya *level alarm high-high* pada awal sistem dimulai. Sehingga dengan menggunakan kontroler tipe PID metode Ziegler-Nichols masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan. Perbandingan respons ketiga kontroler PID metode Ziegler-Nichols dapat dilihat pada Gambar 4.15. dari Gambar 4.15 dapat dilihat bahwa kontroler tipe PID memiliki *overshoot* paling tinggi sebesar 47.65% dan hanya kontroler P saja yang memiliki *offset* sebesar 31%.

Dari respons yang didapat dengan menggunakan metode Ziegler-Nichols berupa kontroler tipe P, PI dan PID, Masih belum dapat mendapatkan respons sesuai dengan spesifikasi desain, karena setiap kontroler menghasilkan respons yang terdapat *overshoot* atau *error steady state*.



Gambar 4.15 Respons Kontroler Metode Ziegler-Nichols

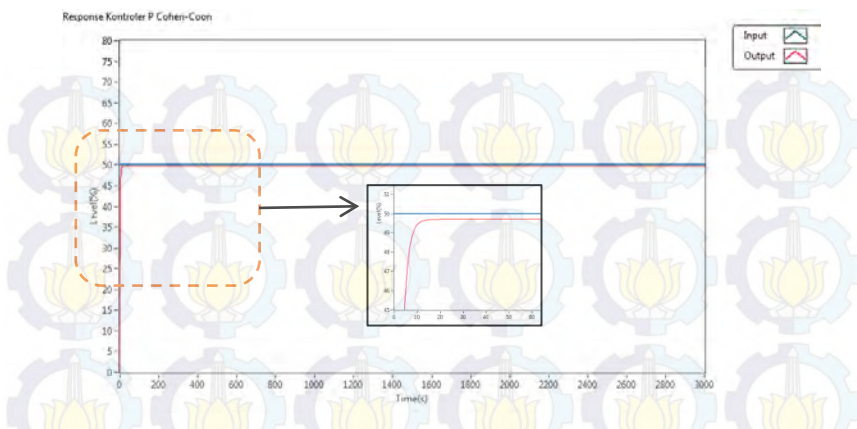
Dalam kondisi *real* level yang terlalu tinggi akan mengaktifkan *level alarm high-high* dimana akan terjadi *shutdown system* dan mengganggu proses lain. Hal ini dapat disebabkan karena metode Ziegler-Nichols merupakan metode yang dilakukan secara grafis, sehingga perbedaan pembacaan grafik dapat mengakibatkan respons berbeda pula. Selain itu, metode ini juga tidak memiliki perhitungan matematis yang sesuai dengan spesifikasi desain yang ditentukan. dan juga kontroler jenis ini memiliki spesifikasi khusus sesuai dengan teori kontroler PID metode Ziegler-Nichols.

4.3.2 Kontroler Metode Cohen-Coon

Dengan memasukkan nilai parameter kontroler PID sesuai dengan metode Cohen-Coon, dapat ditentukan karakteristik respons pada daerah *transient* maupun *steady state* menggunakan kontroler tipe P, PI dan PID.

4.3.2.1 Kontroler Tipe P

Dengan memasukkan parameter kontroler P berupa K_p sebesar 3008.66 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal *step* sebesar 50% sesuai dengan gambar yang ada pada Gambar 4.16.



Gambar 4.16 Respons Kontroler Tipe P Metode Cohen-Coon

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.8.

Tabel 4.8 Karakteristik Kontroler P Metode Cohen-Coon

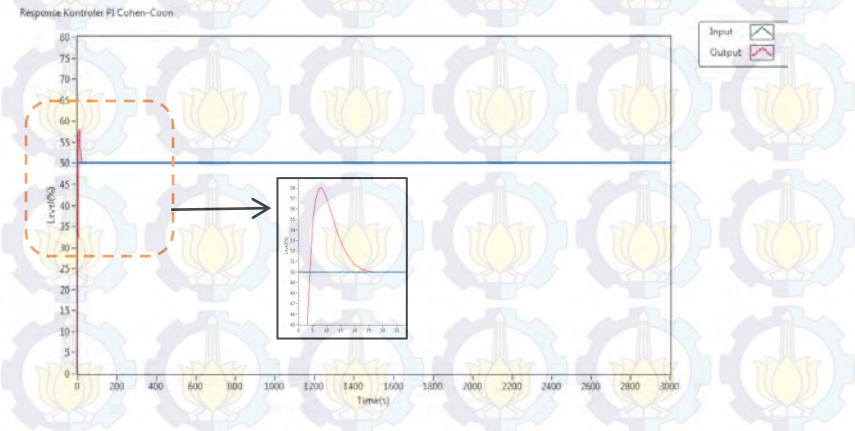
No	Nama	Nilai
1	Y_{SS}	49.675
2	X_{SS}	50.00
3	E_{SS}	0.325
4	$E_{SS}\%$	0.65
5	τ	2.0860
6	K	0.9935
7	$t_r(5\% - 95\%)$	6.1421
8	$t_r(10\% - 90\%)$	4.5834
9	t_d	1.4459
10	$t_s(5\%)$	6.258
11	$t_s(2\%)$	8.344
12	$t_s(0.5\%)$	10.43

Dari respons yang didapat, respons sistem masih memiliki *error steady state* sebesar 0.65%, sehingga dengan menggunakan kontroler

type P metode Cohen-Coon masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan. Sehingga akan dilanjutkan menggunakan kontroler tipe PI.

4.3.2.2 Kontroler Tipe PI

Dengan memasukan parameter kontroler P berupa K_p sebesar 2703.64 serta T_i sebesar 6.288 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal step sebesar 50% sesuai dengan gambar yang ada pada Gambar 4.17.



Gambar 4.17 Respons Kontroler Tipe PI Metode Cohen-Coon

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde dua dan dapat diketahui karakteristik respons seperti pada Tabel 4.9

Tabel 4.9 Karakteristik Kontroler PI Metode Cohen-Coon

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	K	1.00
6	M_p	58.00

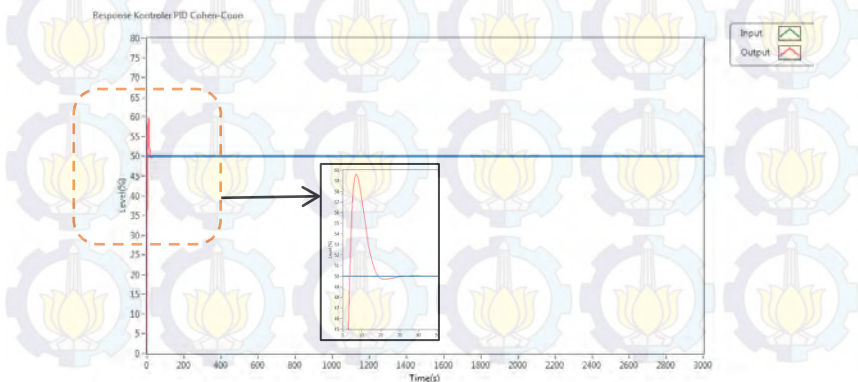
Tabel 4.9 Karakteristik Kontroler PI Metode Cohen-Coon (lanjutan)

No	Nama	Nilai
7	$M_p \%$	16.00
8	t_p	8.022
9	t_d	1.219
10	ζ	0.50406
11	w_n	0.45321
12	τ	4.37743
13	$t_s(5\%)$	13.13229
14	$t_s(2\%)$	17.50973
15	$t_s(0.5\%)$	21.88716

Dari respons yang didapat, dapat dilihat bahwa respons sudah tidak memiliki *error steady state*, namun memiliki *overshoot* sebesar 16% yang dapat mengakibatkan aktifnya *level alarm high-high* pada awal sistem dimulai. Sehingga dengan menggunakan kontroler tipe PI metode Cohen-Coon masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.2.3 Kontroler Tipe PID

Dengan memasukan parameter kontroler PID berupa K_p sebesar 4007.82, T_i sebesar 4.69366 dan T_d sebesar 0.6944 dari tabel yang didapatkan, akan didapatkan respons sistem dengan *input* berupa sinyal step sebesar 50% sesuai dengan gambar yang ada pada Gambar 4.18.



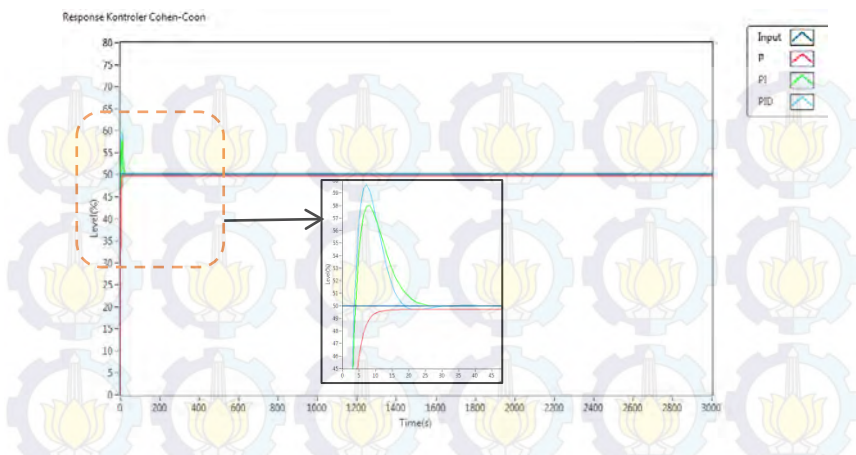
Gambar 4.18 Respons Kontroler Tipe PID Metode Cohen-Coon

Dari respons yang didapat dapat, respons dapat didekatkan ke bentuk orde dua dan dapat diketahui karakteristik respon *transient* dan *steady state* seperti pada Tabel 4.10

Tabel 4.10 Karakteristik Kontroler PID Metode Cohen-Coon

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	K	1.00
6	M_p	59.625
7	$M_p\%$	19.25
8	t_p	7.165
9	t_d	1.307
10	ζ	0.464647995
11	w_n	0.494910999
12	τ	4.348593644
13	$t_s(5\%)$	13.04578093
14	$t_s(2\%)$	17.39437458
15	$t_s(0.5\%)$	21.74296822

Dari respons yang didapat, dapat dilihat bahwa respons sudah tidak memiliki *error steady state*, namun memiliki *overshoot* sebesar 19.25% yang dapat mengakibatkan aktifnya *level alarm high-high* pada awal sistem dimulai. Sehingga dengan menggunakan kontroler tipe PID metode Cohen-Coon masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan. Perbandingan respons ketiga kontroler PID metode Cohen-Coon dapat dilihat pada Gambar 4.19. dari Gambar 4.19 dapat dilihat bahwa kontroler tipe PID memiliki *overshoot* paling tinggi sebesar 19.25% dan hanya kontroler P saja yang memiliki *offset* sebesar 0.65% dan dari ketiga tipe kontroler metode Cohen-Coon memiliki *time constant* lebih cepat dari kontroler dengan metode Ziegler-Nichols. Sehingga untuk kasus ini, kontroler Cohen-Coon lebih cocok digunakan dalam penentuan parameter kontroler pada metode perancangan kontroler secara grafis dibandingkan metode Ziegler-Nichols.



Gambar 4.19 Respons Kontroler Metode Cohen-Coon

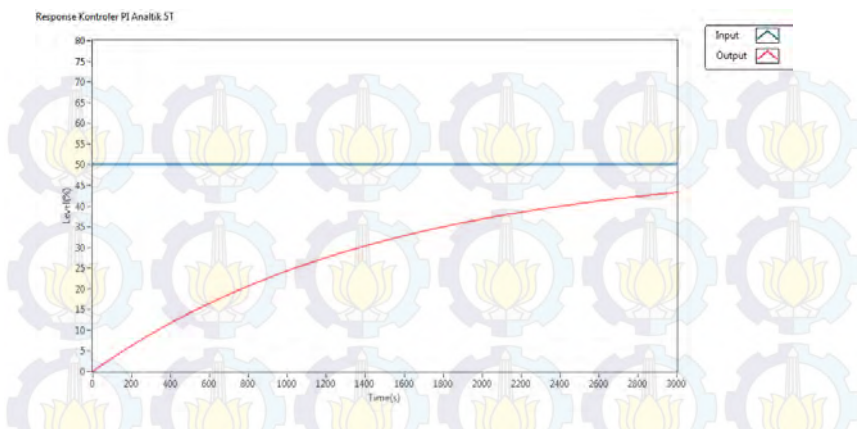
Dari respons yang didapat dengan menggunakan metode Cohen-Coon berupa kontroler tipe P, PI dan PID, Masih belum dapat mendapatkan respons sesuai dengan spesifikasi desain, hal ini dapat disebabkan karena metode Cohen-Coon merupakan metode yang dilakukan secara grafis, sehingga perbedaan pembacaan grafik dapat mengakibatkan respons berbeda pula. Selain itu, metode ini juga tidak memiliki perhitungan matematis yang sesuai dengan spesifikasi desain yang ditentukan, namun memiliki spesifikasi desain sesuai dengan teori perancangan kontroler metode Cohen-Coon sama seperti metode secara grafis Ziegler-Nichols.

4.3.3 Kontroler Metode Analitik

Dengan memasukkan nilai parameter kontroler sesuai dengan metode analitik sesuai perbedaan *time constant*, dapat ditentukan karakteristik respons pada daerah *transient* dan *steady state* sebagai berikut.

4.3.3.1 Kontroler Tipe PI dengan $\tau^* = 5\tau$

Dengan memasukkan parameter kontroler PI berupa K_p sebesar 3.846151 dan T_i sebesar 302.18 dengan *input* berupa sinyal step sebesar 50% didapatkan respons seperti Gambar 4.20.



Gambar 4.20 Respons Kontroler Tipe PI dengan $\tau^* = 5\tau$

Dari respons yang didapat dapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.11

Tabel 4.11 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 5\tau$

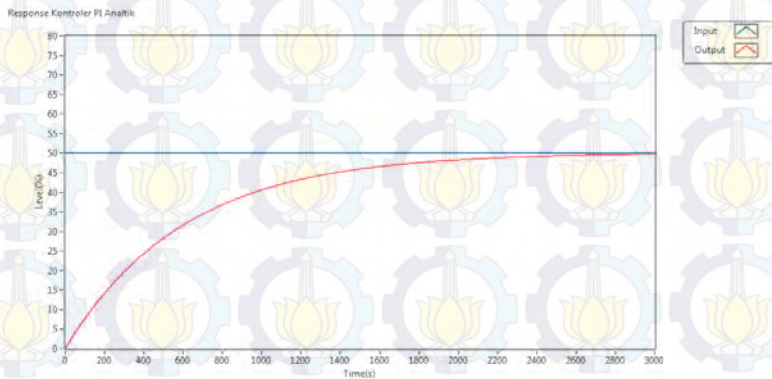
No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	τ	1501.73
6	K	1.00
7	$t_r(5\% - 95\%)$	4421.7523482
8	$t_r(10\% - 90\%)$	3299.6380645
9	t_d	1040.9199155
10	$t_s(5\%)$	4505.19
11	$t_s(2\%)$	6006.92
12	$t_s(0.5\%)$	7508.65

Dari respons yang didapat, respons sistem tidak memiliki *error steady state*, namun untuk respons mencapai kondisi *steady state*

membutuhkan waktu yang besar dengan dapat dilihat nilai *time constan* sebesar 1501.73 atau memiliki *time constant* 2.34 kali lebih besar dari respons sistem *closed loop*, hal ini dapat disebabkan oleh nilai K_p yang terlalu kecil. Dehingga dengan menggunakan kontroler tipe PI metode analitik dengan $\tau^* = 5\tau$ masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.3.2 Kontroler Tipe PI dengan $\tau^* = 2\tau$

Dengan memasukan parameter kontroler PI berupa K_p sebesar 9.615378 dan T_i sebesar 302.18 dengan *input* berupa sinyal step sebesar 50% didapatkan respons seperti Gambar 4.21.



Gambar 4.21 Respons Kontroler Tipe PI dengan $\tau^* = 2\tau$

Dari respons yang didapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.12.

Tabel 4.12 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 2\tau$

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00

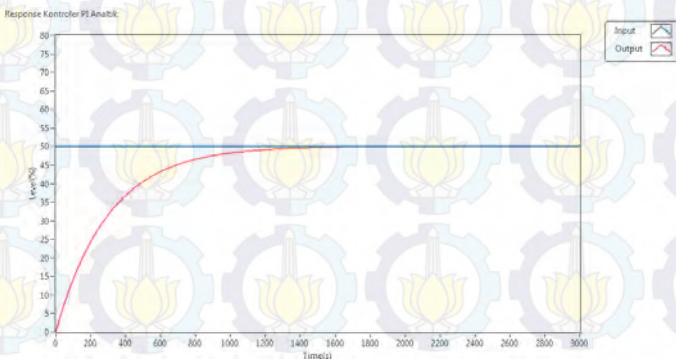
Tabel 4.12 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 2\tau$ (lanjutan)

No	Nama	Nilai
5	τ	600.73
6	K	1.00
7	$t_r(5\% - 95\%)$	1768.812828
8	$t_r(10\% - 90\%)$	1319.9387203
9	t_d	416.3943058
10	$t_s(5\%)$	1802.19
11	$t_s(2\%)$	2402.92
12	$t_s(0.5\%)$	3003.65

Dari respons yang didapat, respons sistem tidak memiliki *error steady state*, respons mencapai kondisi *steady state* membutuhkan waktu yang hampir sama dengan *time constan* sistem *closed loop* sebesar 600.73 atau memiliki *time constant* 0.941 kali dari respons sistem *closed loop*, namun lebih besar 1.989 kali dari *time constant* sistem *open loop*. Hal ini dapat disebabkan oleh nilai K_p yang kecil. Sehingga kontroler tipe PI metode analitik dengan $\tau^* = 2\tau$ masih belum mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.3.3 Kontroler Tipe PI dengan $\tau^* = \tau$

Dengan memasukan parameter kontroler PI berupa K_p sebesar 19.23076 dan T_i sebesar 302.18 dengan *input* berupa sinyal step sebesar 50% didapatkan respons seperti Gambar 4.22.



Gambar 4.22 Respons Kontroler Tipe PI dengan $\tau^* = \tau$

Dari respons yang didapat, respons hasil dari simulasi dapat didekatkan ke dalam bentuk orde satu dan dapat diketahui karakteristik respons pada daerah waktu *transient* dan *steady state* seperti pada Tabel 4.13.

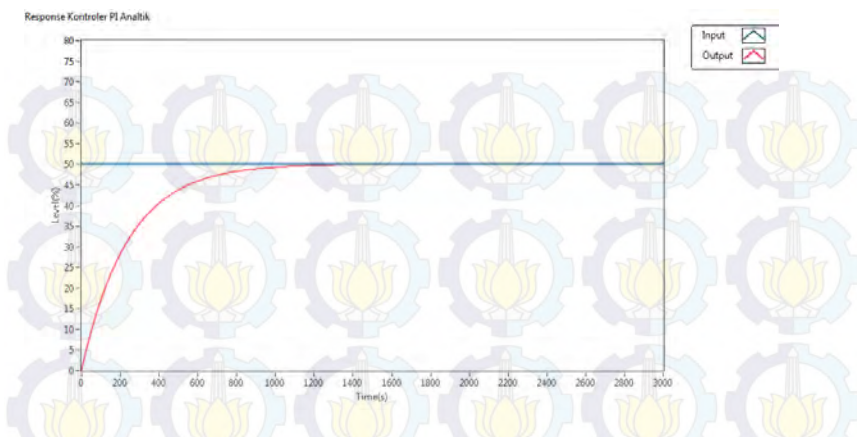
Tabel 4.13 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = \tau$

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	τ	300.73
6	K	1.00
7	$t_r(5\% - 95\%)$	885.4811342
8	$t_r(10\% - 90\%)$	660.7713471
9	t_d	208.4501516
10	$t_s(5\%)$	902.19
11	$t_s(2\%)$	1202.92
12	$t_s(0.5\%)$	1503.65

Dari respons yang didapat, respons sistem tidak memiliki *error steady state*, respons mencapai kondisi *steady state* membutuhkan waktu lebih cepat dengan *time constan* sebesar 300.73 s atau lebih cepat 0.4706 dari sistem *closed loop* dan memiliki *time constant* sama dengan sistem *open loop*, dalam kondisi real dengan memiliki *time constant* yang lebih cepat dapat mempercepat proses pula. Sehingga dengan lebih cepat proses maka akan meningkatkan efektifitas waktu kerja. Hal ini dapat disebabkan oleh nilai T_i yang tepat, sehingga sistem tidak memiliki *offset* dan tidak ada *overshoot*. Sehingga kontroler tipe PI metode analitik dengan $\tau^* = \tau$ sudah mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.3.4 Kontroler Tipe PI dengan $\tau^* = 0.8\tau$

Dengan memasukan parameter kontroler PI berupa K_p sebesar 24.03845 dan T_i sebesar 302.18 dengan *input* berupa sinyal step sebesar 50% didapatkan respons seperti Gambar 4.23.



Gambar 4.23 Respons Kontroler Tipe PI dengan $\tau^* = 0.8\tau$

Dari respons yang didapat dapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada tabel 4.14.

Tabel 4.14 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 0.8\tau$

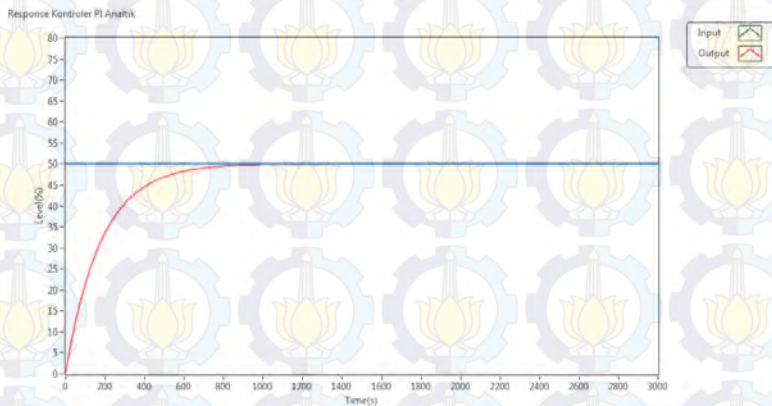
No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00
5	τ	240.73
6	K	1.00
7	$t_r(5\% - 95\%)$	708.8147955
8	$t_r(10\% - 90\%)$	528.9378725
9	t_d	166.8613208
10	$t_s(5\%)$	722.19
11	$t_s(2\%)$	962.92
12	$t_s(0.5\%)$	1203.65

Dari respons yang didapat, respons sistem tidak memiliki *error steady state*, respons mencapai kondisi *steady state* membutuhkan waktu

lebih cepat dengan *time constan* sebesar 240.73 s atau lebih cepat 0.376 dari sistem *closed loop* dan memiliki *time constant* sebesar 0.762 lebih cepat dari sistem *open loop*, Hal ini dapat disebabkan oleh nilai K_p yang besar. Sehingga kontroler tipe PI metode analitik dengan $\tau^* = 0.8\tau$ sudah mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.

4.3.3.5 Kontroler Tipe PI dengan $\tau^* = 0.6\tau$

Dengan memasukkan parameter kontroler PI berupa K_p sebesar 32.05126 dan T_i sebesar 302.18 dengan *input* berupa sinyal step sebesar 50% didapatkan respons seperti Gambar 4.24.



Gambar 4.24 Respons Kontroler Tipe PI dengan $\tau^* = 0.6\tau$

Dari respon yang didapat, respons dapat didekatkan ke bentuk orde satu dan dapat diketahui karakteristik respons seperti pada Tabel 4.15

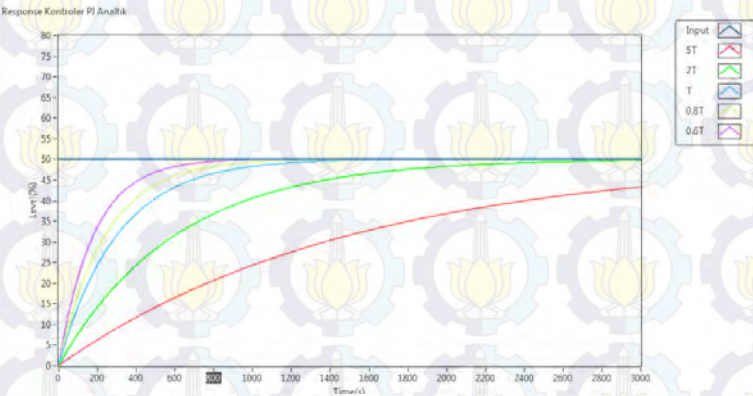
Tabel 4.15 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 0.6\tau$

No	Nama	Nilai
1	Y_{ss}	50.00
2	X_{ss}	50.00
3	E_{ss}	0.00
4	$E_{ss}\%$	0.00

Tabel 4.15 Karakteristik dengan Kontroler PI Metode Analitik $\tau^* = 0.6\tau$ (lanjutan)

No	Nama	Nilai
5	τ	179.73
6	K	1.00
7	$t_r(5\% - 95\%)$	529.204018
8	$t_r(10\% - 90\%)$	394.907173
9	t_d	124.579343
10	$t_s(5\%)$	539.19
11	$t_s(2\%)$	718.92
12	$t_s(0.5\%)$	898.650000

Dari respons yang didapat, respons sistem tidak memiliki *error steady state*, respons mencapai kondisi *steady state* membutuhkan waktu lebih cepat dengan *time constan* sebesar 179.73 atau lebih cepat 0.2813 dari sistem *closed loop* dan memiliki *time constant* sebesar 0.595 lebih cepat dari sistem *open loop*, Hal ini dapat disebabkan oleh nilai K_p yang besar. Sehingga kontroler tipe PI metode analitik dengan $\tau^* = 0.6\tau$ sudah mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan.



Gambar 4.25 Respons Kontroler Tipe PI

Dari perbandingan respons pada Gambar 4.25 yang didapat dengan menggunakan kontroler PI metode analitik dengan dipilih τ^*

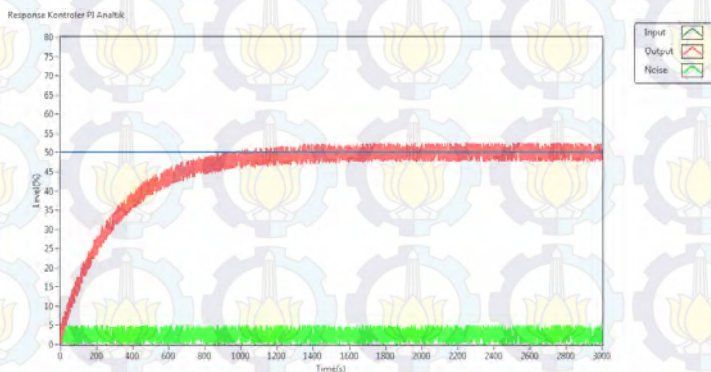
sebesar 5τ , 2τ , τ , 0.8τ dan 0.6τ . Sudah memiliki beberapa respons yang sesuai dengan spesifikasi desain yaitu pada τ^* sebesar τ , 0.8τ dan 0.6τ . yaitu memiliki respons 0% *error steady state*, tidak ada *overshoot* dan *time constant* lebih cepat sistem *open loop* maupun *closed loop*. Hal ini dapat disebabkan perhitungan yang tepat pada nilai T_i sehingga dengan berapapun nilai K_p akan menghasilkan respons yang *zero offset* dan tidak ada *overshoot*. Sementara itu nilai K_p akan mempengaruhi mempercepat atau memperlambat respons. Sehingga parameter kontroler tipe PI metode Analitik dapat digunakan untuk mendapatkan respons sesuai dengan spesifikasi desain yang diinginkan.

4.4 Pengujian Kontroler dengan Gangguan

Pengujian kontroler dengan gangguan dari jenis kontroler yang telah mendapatkan respons sesuai dengan spesifikasi desain yang ditentukan pada bagian sebelumnya. Yaitu menggunakan kontroler PI metode analitik pada τ^* sebesar τ , 0.8τ dan 0.6τ . gangguan yang diberikan menggunakan sinyal *random* sebesar 10% dari nilai *set point* untuk membandingkan pada daerah *transient* dan *steady state* dari sistem tanpa gangguan.

4.4.1 Kontroler Tipe PI $\tau^* = \tau$ dengan Gangguan

Dengan memasukkan parameter kontroler PI berupa K_p dan T_i yang sama dengan bagian sebelumnya, dan ditambah gangguan sebesar 10% maka didapatkan respons seperti pada Gambar 4.26.



Gambar 4.26 Respons Kontroler Tipe PI dengan $\tau^* = \tau$

Dari respons yang didapat, respons dapat diketahui karakteristik respons pada daerah *transient* dan *steady state* seperti pada Tabel 4.16.

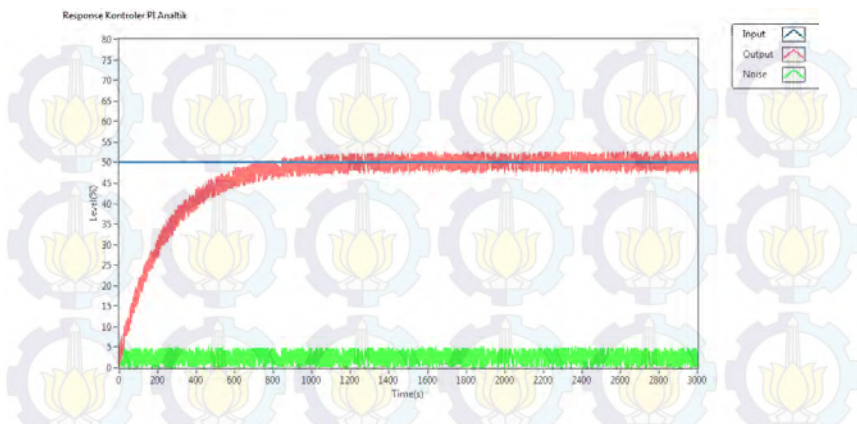
Tabel 4.16 Karakteristik Kontroler PI Metode Analitik $\tau^* = \tau$ dengan Gangguan

No	Nama	Nilai
1	Y_{ssmax}	52.54
2	Y_{ssmin}	47.80
3	$E_{ss\ average}$	1.13
4	$E_{ss\ average}\%$	5.65
5	τ	304.037
6	$t_r(5\% - 95\%)$	895.2183939
7	$t_r(10\% - 90\%)$	668.0375688
8	t_d	210.7423893
9	$t_s(5\%)$	912.1110000
10	$t_s(2\%)$	1216.148
11	$t_s(0.5\%)$	1520.185

Dari respons dengan gangguan didapatkan respons tetap menjaga agar *output* sama dengan *set point* yang diberikan, walaupun tetap memiliki *error* pada daerah *steady state* sebesar 5.65%, dan juga respons menjadi lebih lambat dari respons tanpa gangguan yaitu sebesar 1.01 kali dengan nilai 304.037 s. Dengan bertambahnya *time constant*, maka nilai *time settling* 5%, 2% dan 0.5% ikut bertambah sebesar 1.01 kali sehingga *time settling* 5% bernilai 912.11 s, *time settling* 2% 1216.148 s dan *time settling* 0.5% 1520.185 s. Begitu pula nilai *delay time* dan *rise time* yang ikut bertambah 1.01 kali dari nilai respons tanpa gangguan dengan *rise time* 5%-95% sebesar 896.21 s, *rise time* 10%-90% sebesar 668.047 s dan *delay time* sebesar 210.742 s. Pada *real system* kondisi ini terjadi jika sistem terdapat gangguan pada laju aliran *outlet* atau *inlet* dari *plant test separator* yang bisa diakibatkan juga oleh *control valve*.

4.4.2 Kontroler Tipe PI $\tau^* = 0.8\tau$ dengan Gangguan

Dengan memasukkan parameter kontroler PI berupa K_p dan T_i yang sama dengan bagian sebelumnya, dan ditambah gangguan sebesar 10% maka didapatkan respons seperti pada Gambar 4.27.



Gambar 4.27 Respons Kontroler Tipe PI dengan $\tau^* = 0.8\tau$

Dari respons yang didapat, respons dapat diketahui karakteristik respons pada daerah *transient* dan *steady state* seperti pada Tabel 4.17.

Tabel 4.17 Karakteristik Kontroler PI Metode Analitik $\tau^* = 0.8\tau$ dengan Gangguan

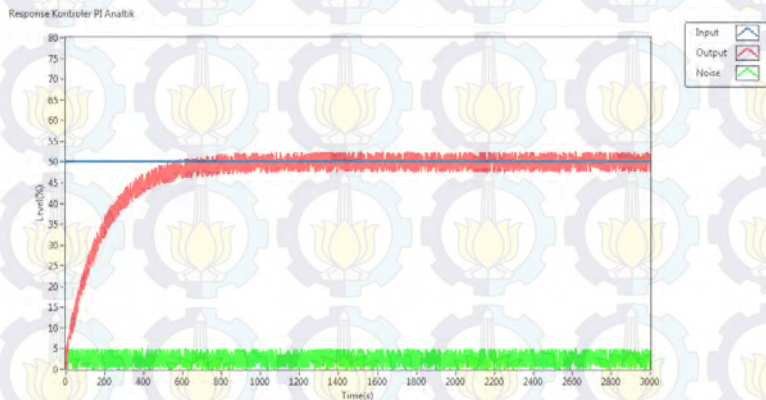
No	Nama	Nilai
1	Y_{SSmax}	52.55
2	X_{SSmin}	47.81
3	$E_{SS\ average}$	1.04
4	$E_{SS\ average}\%$	5.3415511
5	τ	256.58
6	$t_r(5\% - 95\%)$	755.4841533
7	$t_r(10\% - 90\%)$	563.7638821
8	t_d	177.8477036
9	$t_s(5\%)$	769.74
10	$t_s(2\%)$	1026.32
11	$t_s(0.5\%)$	1282.9

Dari respons dengan gangguan didapatkan respons tetap menjaga agar *output* sama dengan *set point* yang diberikan, walaupun tetap

memiliki *error* pada daerah *steady state* sebesar 5.34%, dan juga respons menjadi lebih lambat dari respons tanpa gangguan yaitu sebesar 1.065 kali dengan nilai 256.58 s. Dengan bertambahnya *time constant*, maka nilai *time settling* 5%, 2% dan 0.5% ikut bertambah sebesar 1.065 kali dengan nilai *time settling* 5% sebesar 769.74 s, *time settling* 2% sebesar 1026.32 s dan *time settling* 0.5% sebesar 1282.9 s. Begitu pula nilai *delay time* dan *rise time* yang ikut bertambah 1.065 kali dari nilai respons tanpa gangguan sebesar *rise time* 5%-95% sebesar 755.484 s, *rise time* 10%-90% sebesar 563.763 s dan *delay time* sebesar 177.84 s. seperti pada Tabel 4.15. Pada *real system* kondisi ini terjadi jika sistem terdapat gangguan pada laju aliran *outlet* atau *inlet* dari *plant test separator* yang bisa diakibatkan juga oleh *control valve*.

4.4.3 Kontroler Tipe PI $\tau^* = 0.6\tau$ dengan Gangguan

Dengan memasukan parameter kontroler PI yang sama dengan bagian sebelumnya yaitu K_p sebesar 32.05126 dan T_i sebesar 302.18 , dan ditambah gangguan sebesar 10% maka didapatkan respons seperti pada Gambar 4.28.



Gambar 4.28 Respons Kontroler Tipe PI dengan $\tau^* = 0.6\tau$

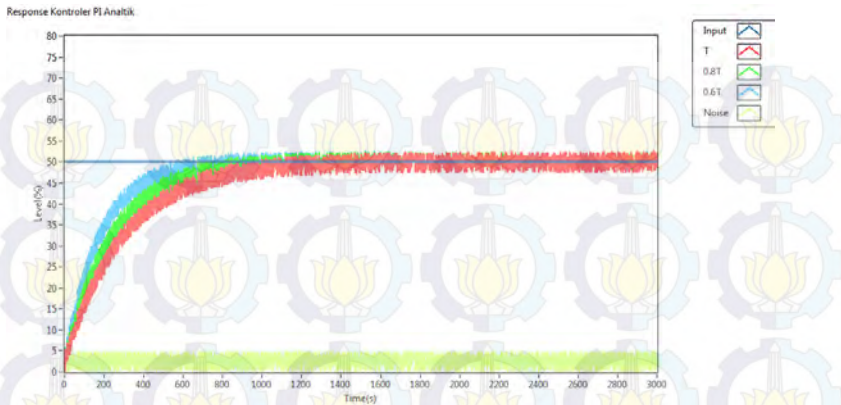
Dari respons yang didapat, respons dapat diketahui perbedaan karakteristik respons pada daerah *transient* dan *steady state* antara sistem dengan gangguan dan tanpa gangguan seperti pada Tabel 4.18.

Tabel 4.18 Karakteristik Kontroler PI Metode Analitik $\tau^* = 0.6\tau$ dengan Gangguan

No	Nama	Nilai
1	Y_{SSmax}	52.57
2	X_{SSmin}	47.83
3	$E_{SS\ average}$	1.01
4	$E_{SS\ average}\%$	5.179487
5	τ	186.304
6	$t_r(5\% - 95\%)$	548.56076
7	$t_r(10\% - 90\%)$	409.351728
8	t_d	129.136092
9	$t_s(5\%)$	558.912
10	$t_s(2\%)$	745.216
11	$t_s(0.5\%)$	931.52

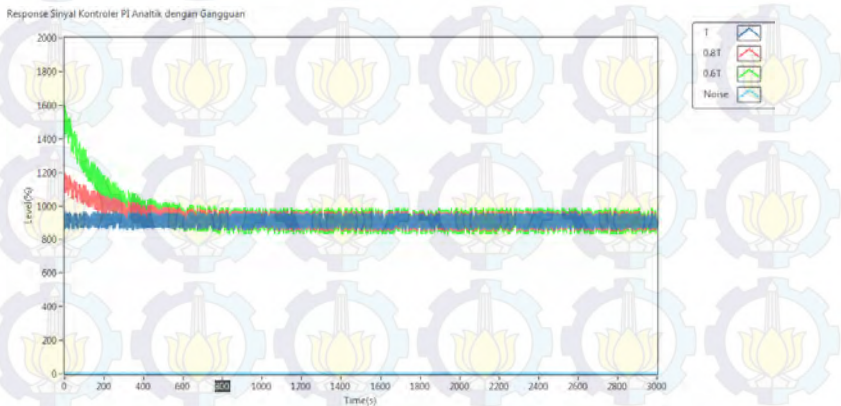
Dari respons dengan gangguan didapatkan respons tetap menjaga agar *output* sama dengan *set point* yang diberikan, walaupun tetap memiliki *error* pada daerah *steady state* sebesar 5.179%, dan juga respons menjadi lebih lambat dari respons tanpa gangguan yaitu sebesar 1.036 kali bernilai 186.304. Dengan bertambahnya *time constant*, maka nilai *time settling* 5%, 2% dan 0.5% ikut bertambah sebesar 1.036 kali yaitu *time settling* 5% sebesar 558.912 s, *time settling* 2% sebesar 745.216 s dan *time settling* 0.5% sebesar 931.52 s. Begitu pula nilai *delay time* dan *rise time* yang ikut bertambah 1.036 kali dari nilai respons tanpa gangguan sebesar *rise time* 5%-95% sebesar 548.56 s, *rise time* 10%-90% sebesar 409.351 s dan *delay time* sebesar 129.136092 s.

Dari respons yang didapat dengan menggunakan kontroler PI metode analitik dengan dipilih τ^* sebesar τ , 0.8τ dan 0.6τ diberi gangguan. Respons ketiga kontroler memiliki *error steady state* sekitar 5% dan memiliki *time constan* lebih lambat sebesar 1.03 kali dari respons sebelumnya. Dengan bertambahnya nilai *time constant* mengakibatkan nilai *time rise*, *time delay* dan *time settling*. Namun dari ketiga kontroler semua *output* sistem menuju pada nilai *set point*. Sehingga dalam kondisi *real system* nilai PV akan berusaha sesuai dengan SP walaupun terdapat gangguan pada sistem. Perbandingan respons kontroler PI dengan diberikan gangguan seperti pada Gambar 4.29.



Gambar 4.29 Respons Kontroler Tipe PI dengan Gangguan

Selanjutnya sinyal kontrol dari kontroler akan diamati yang berguna untuk memilih kontroler mana yang memiliki sinyal kontrol yang tidak fluktuatif, karena fluktuatifnya sinyal kontrol berakibat pada fluktuatifnya juga aktuator yang digunakan, pada kasus ini aktuator berupa *control valve*. Jika *control valve* lebih sering terjadi proses buka-tutup akan mengakibatkan *control valve* menjadi lebih cepat rusak. Sehingga dari pengamatan sinyal kontrol akan dilihat kontroler mana yang memiliki sinyal kontrol paling rendah nilai fluktuatifnya.



Gambar 4.30 Respons Sinyal Kontrol Tipe PI dengan Gangguan

Dari Gambar 4.30 dapat ditentukan karakteristik sinyal kontrol dari ketiga kontroler seperti pada Tabel 4.17.

Tabel 4.19 Karakteristik Sinyak Kontrol dengan Gangguan

No	Tipe	u_{Max}	u_{Min}	% $u_{Amplitude}$
1	$\tau^* = \tau$	956.9	860.9	10.0323
2	$\tau^* = 0.8\tau$	970.3	849.3	12.4703
3	$\tau^* = 0.6\tau$	989.9	829.5	16.2036

Dari pengamatan sinyal kontrol dengan gangguan seperti pada Gambar 4.30 dan Tabel 4.17, sistem dengan *time constant* lebih cepat memiliki sinyal kontrol yang cenderung lebih besar dan lebih fluktuatif dibandingkan dengan sinyal kontrol *time constant* lebih kecil. Yaitu kontroler dengan $\tau^* = 0.6\tau$ memiliki nilai amplitude perubahan sinyal kontrol sebesar 16.2036% dan paling rendah kontroler dengan $\tau^* = \tau$ memiliki nilai 10.0323. Besarnya nilai fluktuatif dari sinyal kontrol tidak baik untuk aktuator yang digunakan, pada kasus ini aktuator yang digunakan berupa *control valve*. Terlalu fluktuatifnya sinyal kontrol dapat mengakibatkan *control valve* mudah rusak.



BAB V

PENUTUP

5.1 Kesimpulan

Setelah melakukan simulasi, pengujian, dan analisa, dapat diambil beberapa kesimpulan, antara lain:

1. Kontroler PID metode Ziegler-Nichols belum bisa memenuhi spesifikasi desain yang telah ditentukan, Karena kontroler tipe P memiliki *error steady state* sebesar 31.00%, kontroler tipe PI memiliki *overshoot* sebesar 39.50% dan kontroler tipe PID memiliki *overshoot* sebesar 47.65%.
2. Kontroler PID metode Cohen-Coon belum bisa memenuhi spesifikasi desain yang telah ditentukan, Karena kontroler tipe P memiliki *error steady state* sebesar 0.65%, kontroler tipe PI memiliki *overshoot* sebesar 16.00% dan kontroler tipe PID memiliki *overshoot* sebesar 19.25%.
3. Kontroler tipe analitik $\tau^* = \tau$ dengan parameter kontroler $K_p = 19.23076$ dan $T_i = 302.18$ s memenuhi spesifikasi desain yang telah ditentukan, pengujian tanpa gangguan memiliki nilai $E_{ss} \% = 0\%$, $\tau = 300.73$ s, $T_r(5\%-95\%) = 885.48$ s, $T_d = 208.45$ s, $T_s(5\%) = 902.19$ s, dan pengujian dengan gangguan didapatkan $E_{ss} \% = 5.65\%$ dan $\tau = 304.73$ s serta besarnya fluktuatif dari sinyal kontrol sebesar 10.0323%.

5.2 Saran

Untuk penelitian selanjutnya dapat dilakukan dengan melakukan pengaturan *variable* lain dari *test separator* berupa *pressure* gas atau *level* air. Serta dapat melakukan penelitian mengenai pengaturan dalam model *cascade* untuk memperpanjang usia *control valve*.

DAFTAR PUSTAKA

- [1] Nusantara, D., "Perancangan Sistem Pengendalian Level pada Monitoring Produksi Sumur Minyak dan Gas dengan Menggunakan Kontroler PID di PT Pertamina EP Region Jawa, Field Subang-Tambun", *Tugas Akhir Teknik Fisika FTI-ITS*, Surabaya: JTF-FTI-ITS, 2010.
- [2] Fauzan, R., "Well Test Comparison TGTS and TBRG on SPU Total E&P Indonesia", *Laporan Magang Teknik Fisika FTI-ITB*. Bandung: JTF-FTI-ITB, 2015.
- [3] Shayda, Atalla F. dan James H. Taylor, "Modeling and Control of Three-Phase Gravity Separators in Oil Production Facilities", *Proc. of American Control Conference*, pp. 4847-4853, USA, Juli. 2007.
- [4] Taufiq, M., "Perancangan Ulang Pengendalian Ketinggian Minyak dengan Pengendali PI Berbasis Jaringan Syaraf Tiruan (JST) pada Separator Tiga Fasa di Unit TPA Oil Train Total E&P Indonesia", *Tugas Akhir Teknik Fisika FTI-ITS*, Surabaya: JTF-FTI-ITS, 2010.
- [5], "*Intervention Report Level Transmitter – Test Separator WPS2*", Instrumentation Maintenance Department Total E&P Indonesia, 2015
- [6] Vocational Training, "**Introduction to Oil and Gas Production**", American Petroleum Institute, 1996.
- [7] Rasyid, M. A. dan D. Yamanda., "Studi Process Control System (PCS), Instrumentation Control System (ICS) dan Safety Shutdown System (SSS) pada Separator Tiga Fasa V-601 di DC Kuat Gas Plant Energi Mega Persada Malacca Strait S.A.", *Kerja Praktek Teknik Elektro FTI-ITS*, Surabaya: JTE-FTI-ITS, 2015.
- [8] Engineering, Construction and Project, "**WPS2 Platform Test Separator Piping and Instrumentation Diagram – 32VZ1101**", Total E&P Indonesia, 2007.
- [9] Engineering, Construction and Project, "**PJC_SN2B&PCK7B_EPSC(1+2)_PHI_WPS2 Process & Utility Operating and Control Philosophy**", Total E&P Indonesia, 2014.
- [10] Engineering, Construction and Project, "**Training Manual Exploration and Production Equipment - Separator**", TOTAL,

2007

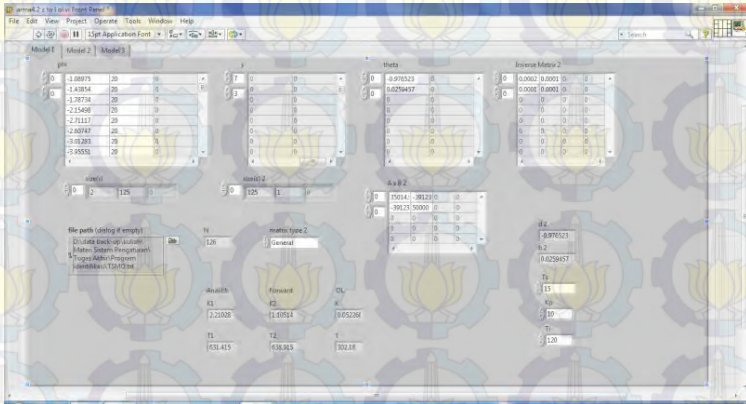
- [11] Oil and Gas Course, "**SEPARATORS**", IFP Training, 2013
- [12] Johnson, C., "*Process Control Instrumentation Technology*", New Jersey: Prentice Hall, 1993.
- [13], "*Instrumentation Symbols and Identification – ANSI/ISA-5.1-1984 (R1992)*", ISA-The Instrumentation, System, and Automation Society, 1992.
- [14] Ljunt, L. dan T. Gland, "*Modeling of Dynamics Systems*", New Jersey: Prentice Hall, 1994.
- [15] Iskandar, E., "*Sistem Pengaturan Adaptif*," Lab TSP. JTE-FTI-ITS, Surabaya, 2015.
- [16] Ogata, K., "*Discrete-Time Control System*", New Jersey: Prentice Hall, 1998.
- [17] Ogunnaike, B. A. dan W. H. Ray, "*Process Dynamics, Modelling, and Control*", New York: Oxford University Press, 1994.
- [18] Lee, S. W. S. J. dan I.-B. Lee, "*Process Identification and PID Control*", Singapore: John Wiley & Sons (Asia) Pte Ltd, 2009.
- [19] Nise, N., "*Control System Engineering*", United States of America: John Wiley & Sons, 2011.
- [20] Ogata, K., "*Modern Control Engineering*", New Jersey: Prentice Hall, 2010.
- [21] Maintenance - Instrumentation Control System, "**Training Manual The Regulator and Its Function**", Total E&P Indonesia, 2008
- [22] Ziegler, J.G. dan N. B. Nichols, "Optimum Setting for Automatic Controllers.", *Transaction of The A.S.M.E*, pp. 759-1768, New York, November. 1942.
- [23] Gamayanti, N., "*Diktat Mata Kuliah Dasar Sistem Pengaturan*," Lab TSP. JTE-FTI-ITS, Surabaya, 2005.
- [24] Engineering, Construction and Project, "**PJC_SN2B&PCK7B_EPSC(1+2)_DTL_WPS2 PCS MIMIC DISPLAY**", Total E&P Indonesia, 2014.
- [25] Nugroho, B. G., "Pengaturan Kecepatan Motor Induksi Tiga Fasa pada Mesin Sentrifugal Menggunakan Pengendali Anti-Windup PID dengan Prediksi Integral State", *Tugas Akhir Teknik Elektro FTI-ITS*. Surabaya: JTE-FTI-ITS, 2015.

- [26] Pratama, M., “Studi Perbandingan Metode Anti *Wind Up* Konvensional dan Fuzzy pada Sistem Pengaturan Kecepatan Motor Servo DC”, *Tugas Akhir Teknik Elektro FTI-ITS*. Surabaya: JTE-FTI-ITS, 2010.
- [27] Yhang, Zhenyu dan M. Juhl, "On the Innovation of Level Control of an Offshore Three-Phase Separator," *Pro. of International Conference on Mechatronics and Automation*, pp. 1348-1353, China, Agustus. 2010.

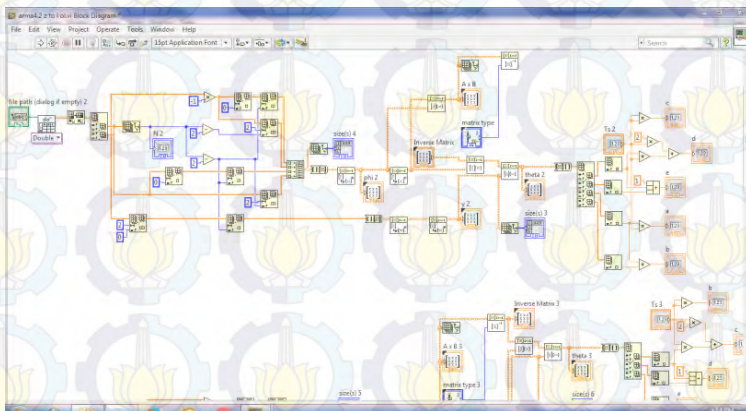


LAMPIRAN

A. Program Identifikasi Sistem

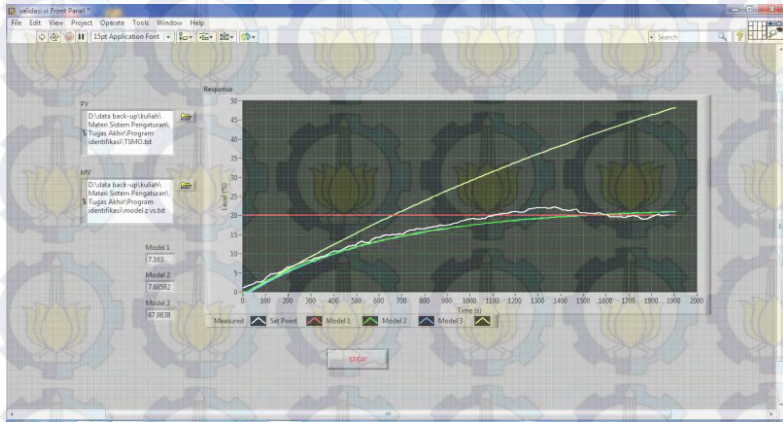


Tampilan *Front Panel* Program Identifikasi Sistem

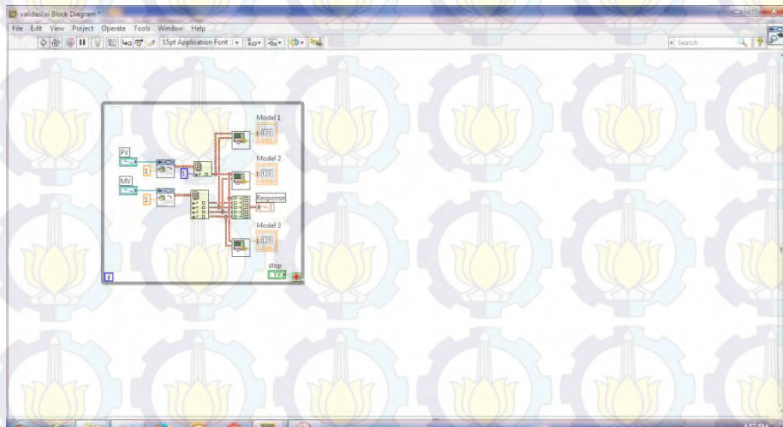


Tampilan *Block Diagram* Program Identifikasi Sistem

B. Program Validasi Model

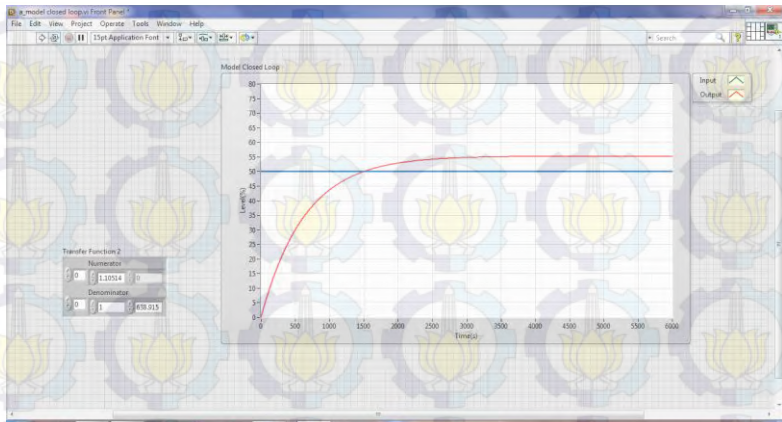


Tampilan *Front Panel* Program Validasi Model

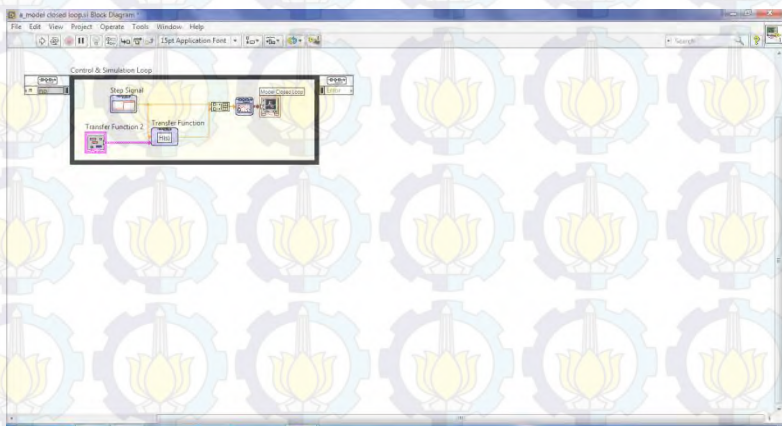


Tampilan *Block Diagram* Program Validasi Model

C. Program Uji *Closed Loop*

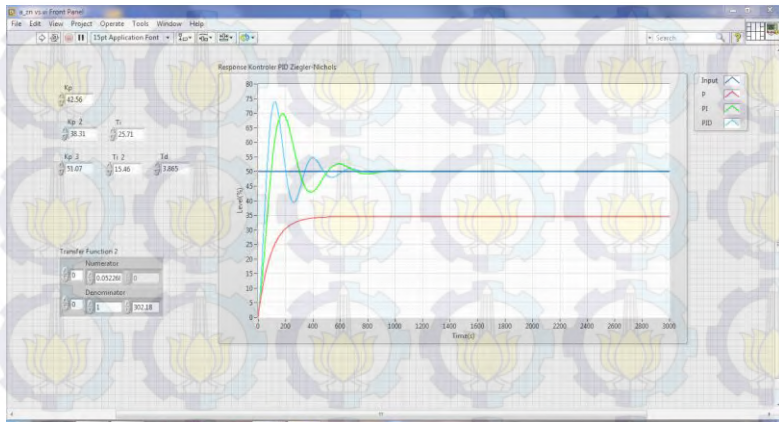


Tampilan *Front Panel* Program Uji *Closed Loop*

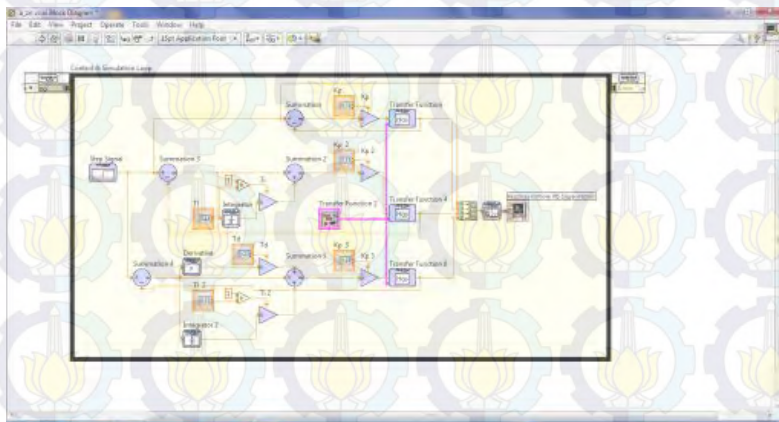


Tampilan *Block Diagram* Program Uji *Closed Loop*

D. Program Kontroler Ziegler-Nichols

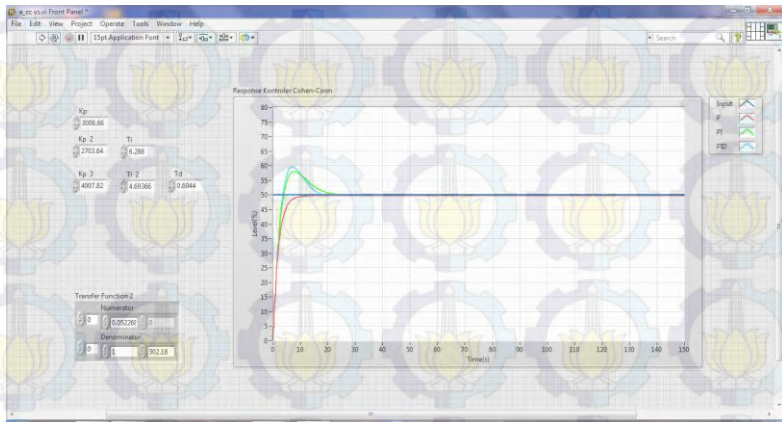


Tampilan Front Panel Program Kontroler Ziegler-Nichols

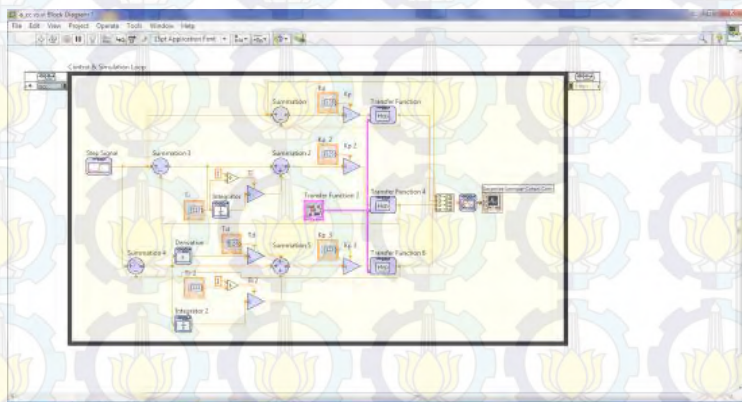


Tampilan Block Diagram Program Kontroler Ziegler-Nichols

E. Program Kontroler Cohen-Coon

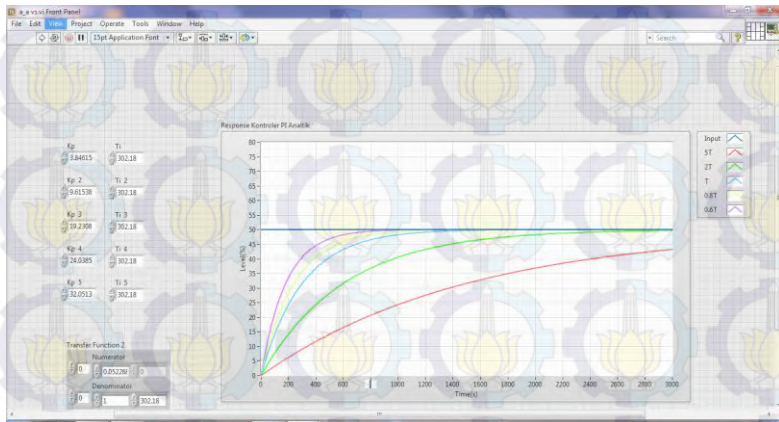


Tampilan *Front Panel* Program Kontroler Cohen-Coon

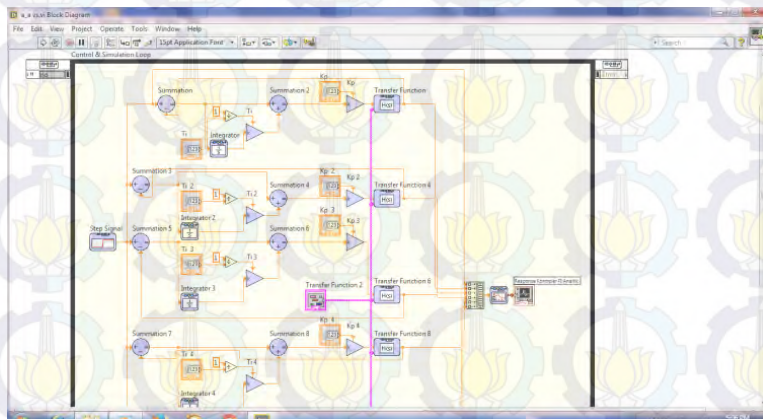


Tampilan *Block Diagram* Program Kontroler Cohen-Coon

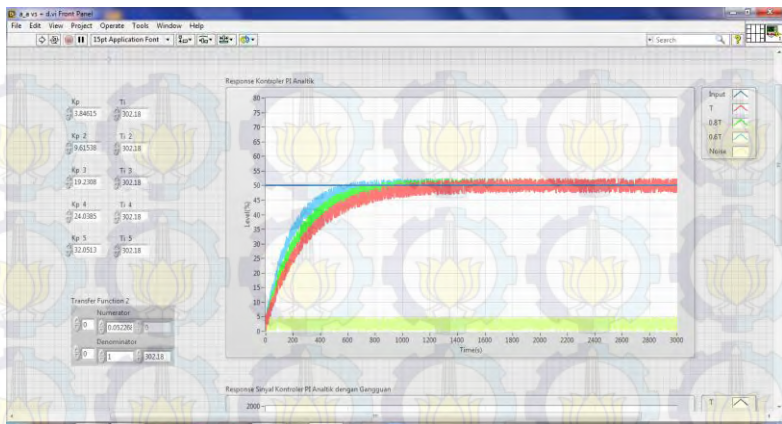
F. Program Kontroler Analitik



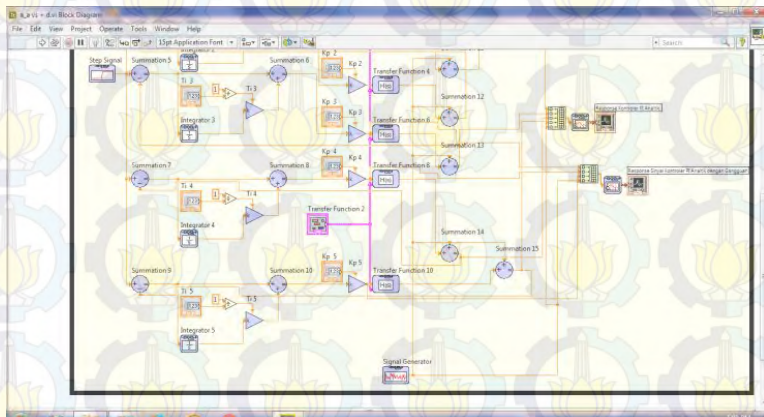
Tampilan *Front Panel* Program Kontroler Analitik



Tampilan *Block Diagram* Program Kontroler Analitik

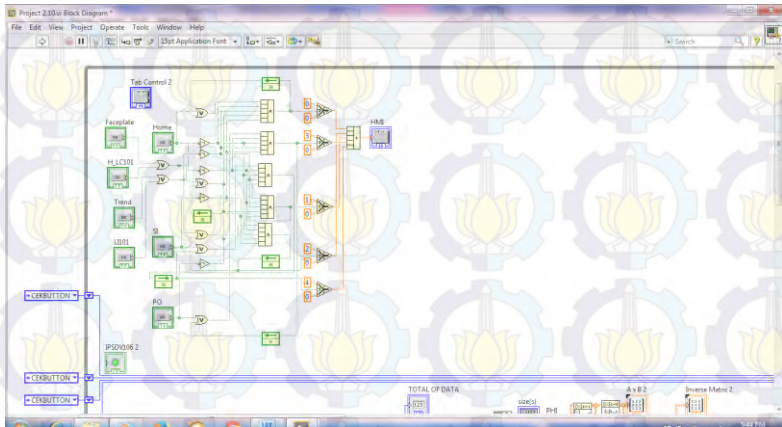


Tampilan *Front Panel* Program Kontroler Analitik dengan - Gangguan

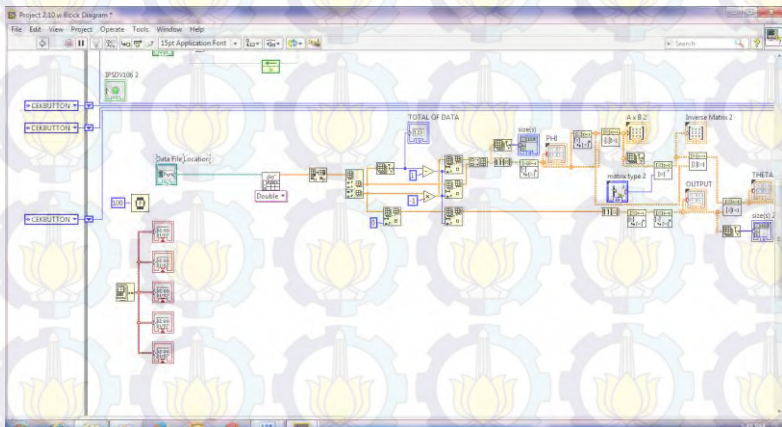


Tampilan *Block Diagram* Program Kontroler Analitik dengan - Gangguan

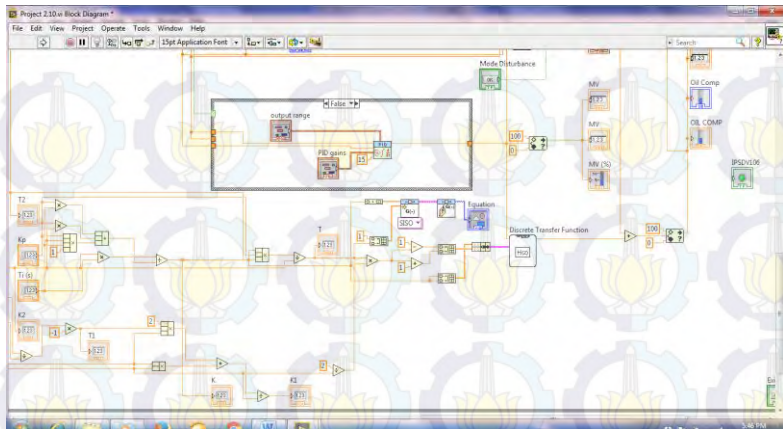
G. Program Simulator HMI



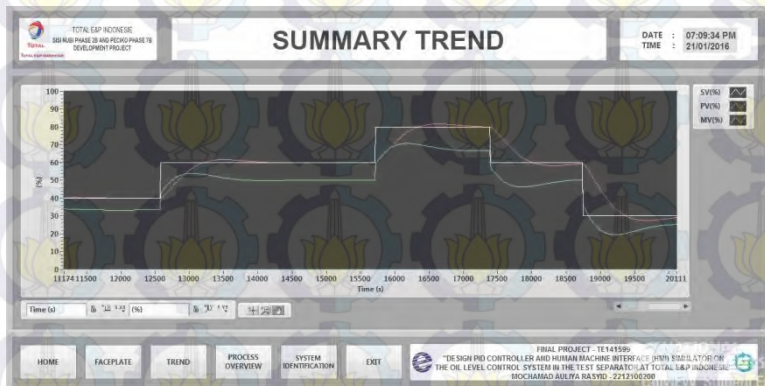
Tampilan *Block Diagram* Program Simulator HMI (1)



Tampilan *Block Diagram* Program Simulator HMI (2)



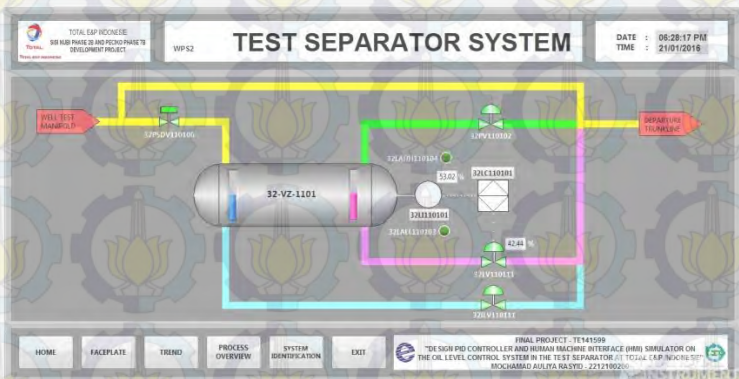
Tampilan *Block Diagram* Program Simulator HMI (5)



Tampilan *Window Trend*



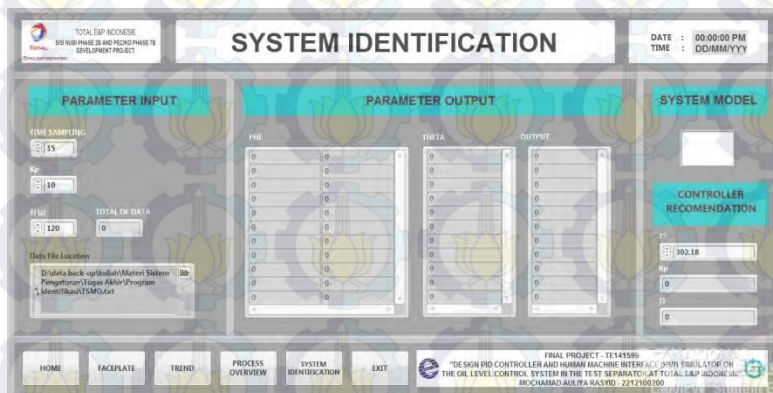
Tampilan Window Process Overview



Tampilan Window Home

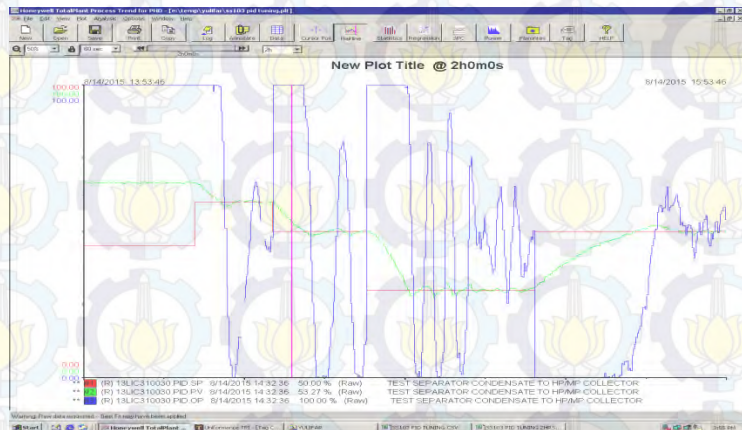


Tampilan Window Faceplate

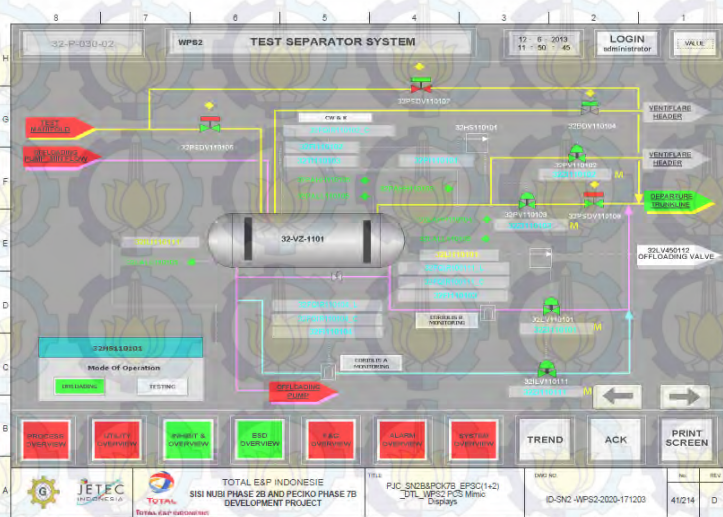


Tampilan Window System Identification

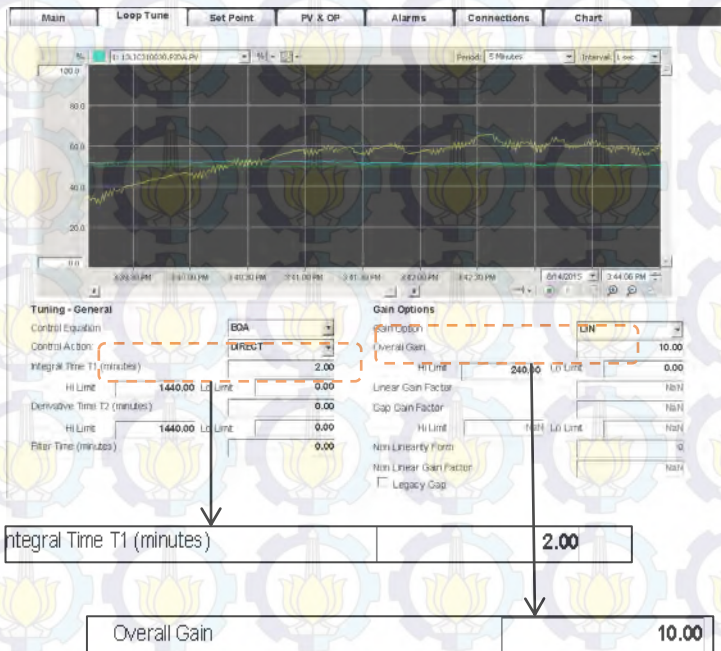
H. Tampilan HMI Real System



Tampilan PHD Real System

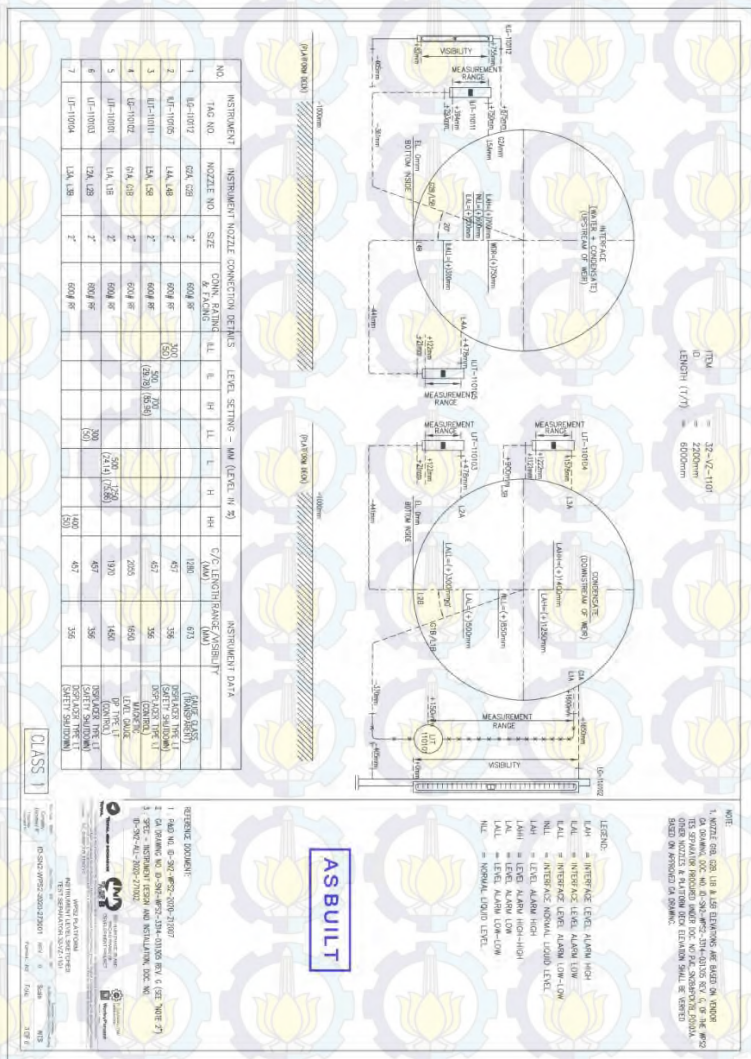


Tampilan HMI Real System



Tampilan Faceplate Real System

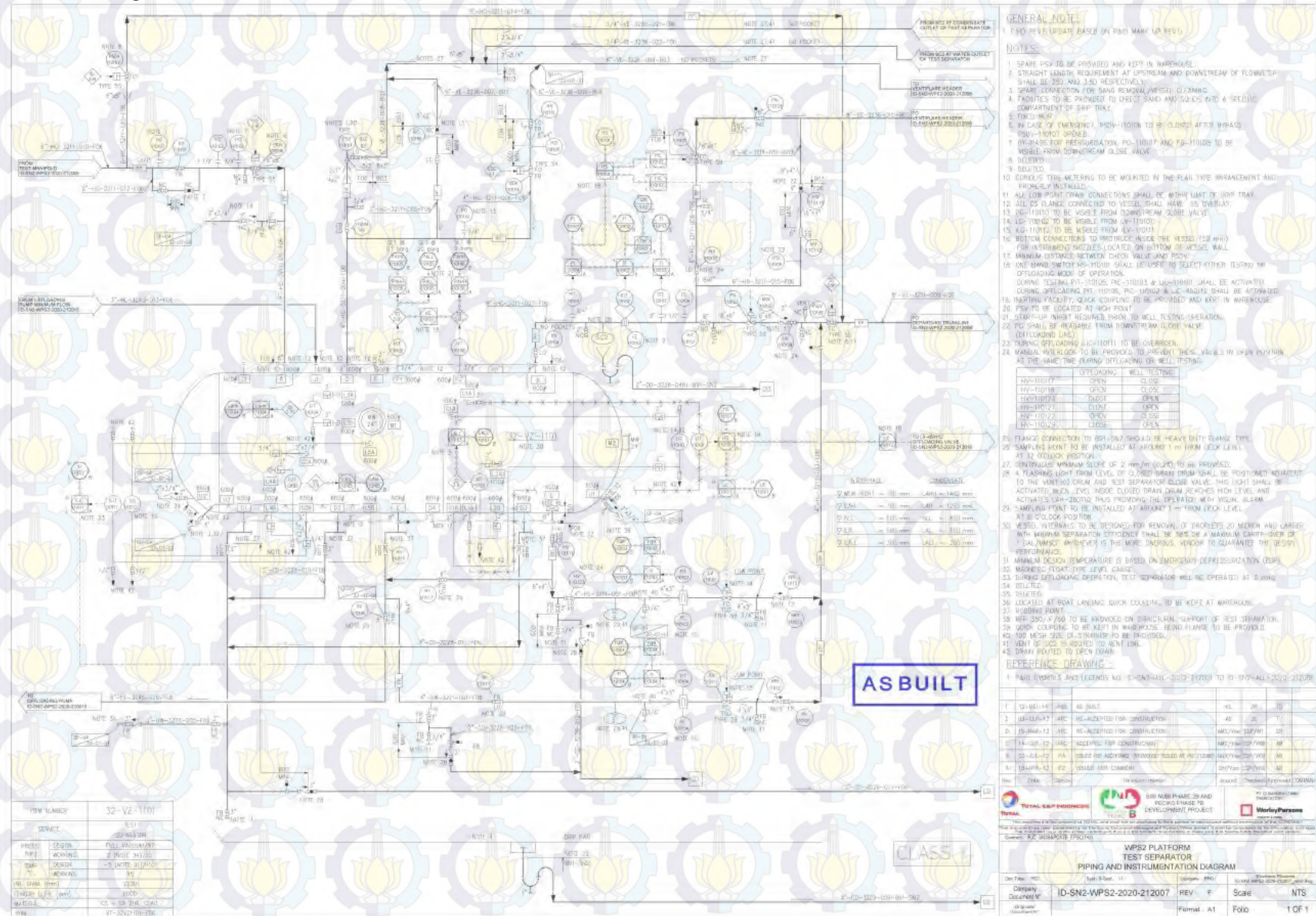
I. *Level Sketch* Test Separator



J. Data Sheet Control Valve

GENERAL	1	Tag No.	32-LV -110101	
	2	Service	TEST SEPARATOR LEVEL (CONDENSATE SIDE)	
	3	P&ID No.	Line No.	ID-SN2-WPS2-2020-212007
	4	Piping Class	Valve Service Class	4"-HL-3211-001-F06
	5	Hazardous Area	Electrical Certification	F06
PROCESS CONDITION	6	Ingress Protection	Zone 1, Group IIA, Temp. Class T3	
	7		Units	Case 1 *
	8	Fluid Type	-	Case 2 *
	9	Fluid Phase	-	Case 3 *
	10	Corrosive/Erosive Comp.	-	Case 4 *
	11	Flow Rate	-	Case 5 *
	12	Pressure Operating	-	Case 6 *
	13	Pressure Drop	-	
	14	Temperature Operating	-	
	15	MW	-	
	16	Compressibility Factor	-	
	17	Density @ FTP	-	
	18	Viscosity	-	
	19	Vapour Pressure	-	
	20	Critical Pressure	-	
LINE	21	Cv Calc	-	
	22	Opening	-	
	23	Flow Status	-	
	24	Predicted SPL	-	
	25		-	
	26	Pipe Material	-	
	27	Pipe Size	-	
	28	Pipe Sch.	-	
	29	Pipe Insulation	-	
	30	Design Press	-	
	31	Design Temp.	-	
	32	Body Type	-	
	33	ANSI Class	-	
	34	Size Inlet	-	
	35	Flange Face Finish	-	
BODY AND TRIM	36	Body Size	-	
	37	Guiding	-	
	38	Flow Direction (up/down)	-	
	39	Bonnet Type	-	
	40	Lubricator	-	
	41	Body / Bonnet Material	-	
	42	Trim Characteristic	-	
	43	Trim Style	-	
	44	Balanced/Unbalanced	-	
	45	Valve Cv	-	
	46	Plug/ Ball/ Disk Material	-	
	47	Cage/Guide Material	-	
	48	Seat Material	-	
	49	Bearing Material	-	
	ACTUATOR	50	Shaft Material	-
51		Packing Material	-	
52		Shut-off Class	-	
53		Max. SPL Allowable	-	
54		Manufacturer	-	
55		Actuator Type	-	
56		On/Off or Modulating	-	
57		Actuator Material	-	
58		Fluid Supply	-	
59		Supply Min / Norm. Des.	-	
60		Fail Position	-	
61		Close at	-	
62		Max. Shut-off Diff. Press.	-	
63		Stroking Time	-	
POSITIONER		64	Fluid Supply Connection	-
	65	Manufacturer / Model No.	-	
	66	Positioner Tag No.	-	
	67	Positioner Type	-	
	68	Direct / Reverse Acting	-	
	69	Supply/In/Out Gauges	-	
	70	Manual Bypass	-	
	71	Transducer Mounting	-	
	72	Input Signal/Output Signal	-	
	73	Electrical Connection	-	
	74	Manufacturer / Model No.	-	
	75	Tubings	-	
	76	Handwheel	-	
	77	Limit Switch	-	
	ACCESSORIES	78	Mechanical Position Ind.	-
79		Lock-Up Devices	-	
80		Function Test	-	
81		Seat Leakage	-	
82		Material Cert.	-	
83		Painting	-	
84			-	
85			-	
86			-	
87			-	
88			-	
89			-	
90			-	
91			-	
TEST/ CERT.		92		-
	93		-	
	94		-	
	95		-	
	96		-	
	97		-	
	98		-	
	99		-	
	100		-	
	101		-	
	102		-	
	103		-	
	104		-	
	105		-	
	NOTES	106		-
107			-	
108			-	
109			-	
110			-	
111			-	
112			-	
113			-	
114			-	
115			-	
116			-	
117			-	
118			-	
119			-	
120			-	
FAR	121		-	
	122		-	
	123		-	
	124		-	
	125		-	
	126		-	
	127		-	
	128		-	
	129		-	
	130		-	
	131		-	
	132		-	
	133		-	
	134		-	
	MIRI	135		-
136			-	
137			-	
138			-	
139			-	
140			-	
141			-	
142			-	
143			-	
144			-	
145			-	
146			-	
147			-	
148			-	
CIRI		149		-
	150		-	
	151		-	
	152		-	
	153		-	
	154		-	
	155		-	
	156		-	
	157		-	
	158		-	
	159		-	
	160		-	
	161		-	
	162		-	
	BIRI	163		-
164			-	
165			-	
166			-	
167			-	
168			-	
169			-	
170			-	
171			-	
172			-	
173			-	
174			-	
175			-	
176			-	
177			-	
178		-		
179		-		
180		-		
181		-		
182		-		
183		-		
184		-		
185		-		
186		-		
187		-		
188		-		
189		-		
190		-		
191		-		
192		-		
193		-		
194		-		
195		-		
196		-		
197		-		
198		-		
199		-		
200		-		
201		-		
202		-		
203		-		
204		-		
205		-		
206		-		
207		-		
208		-		
209		-		
210		-		
211		-		
212		-		
213		-		
214		-		
215		-		
216		-		
217		-		
218		-		
219		-		
220		-		
221		-		
222		-		
223		-		
224		-		
225		-		
226		-		
227		-		
228		-		
229		-		
230		-		
231		-		
232		-		
233		-		
234		-		
235		-		
236		-		
237		-		
238		-		
239		-		
240		-		
241		-		
242		-		
243		-		
244		-		
245		-		
246		-		
247		-		
248		-		
249		-		
250		-		
251		-		
252		-		
253		-		
254		-		
255		-		
256		-		
257		-		
258		-		
259		-		
260		-		
261		-		
262		-		
263		-		
264		-		
265		-		
266		-		
267		-		
268		-		
269		-		
270		-		
271		-		
272		-		
273		-		
274		-		
275		-		
276		-		
277		-		
278		-		
279		-		
280		-		
281		-		
282		-		
283		-		
284		-		
285		-		
286		-		
287		-		
288		-		
289		-		
290		-		
291		-		
292		-		
293		-		
294		-		
295		-		
296		-		
297		-		
298		-		
299		-		
300		-		
301		-		
302		-		
303		-		
304		-		
305		-		
306		-		
307		-		
308		-		
309		-		
310		-		
311		-		
312		-		
313		-		
314		-		
315		-		
316		-		
317		-		
318		-		
319		-		
320		-		
321		-		
322		-		
323		-		
324		-		
325		-		
326		-		
327		-		
328		-		
329		-		
330		-		
331		-		
332		-		
333		-		
334		-		
335		-		
336		-		
337		-		
338		-		
339		-		
340		-		
341		-		
342		-		
343		-		
344		-		
345		-		
346		-		
347		-		
348		-		
349		-		
350		-		
351		-		
352		-		
353		-		
354		-		
355		-		
356		-		
357		-		
358		-		
359		-		
360		-		
361		-		
362		-		
363		-		
364		-		
365		-		
366		-		
367		-		
368		-		
369		-		
370		-		

K. P&ID Test Separator



RIWAYAT PENULIS



Mochamad Auliya Rasyid atau yang biasa dipanggil Auliya/ Oya lahir di Bogor, 12 Desember 1993. Ia merupakan anak pertama dari empat bersaudara dari pasangan Bapak Muhamad Ramdani dan Ibu Nursobah. Tinggal bersama orang tua di Bukit Cimanggu City Blok K1 No.2 Kota Bogor, Jawa Barat Indonesia. Penulis memulai pendidikannya di SD Bina Insani Kota Bogor, dilanjutkan di SMPN 5 Kota Bogor, dan SMAN 1 Kota Bogor. Setelah lulus, pada tahun 2012 melanjutkan pendidikan jenjang sarjana sebagai mahasiswa Regular S1 Teknik Elektro ITS Surabaya, berkonsentrasi pada Bidang Studi Teknik Sistem Pengaturan. Pada masa perkuliahan, penulis aktif dalam kegiatan organisasi mahasiswa di Departement Hubungan Luar Himpunan Mahasiswa Teknik Elektro (Himatektro) dan asisten Laboratorium Teknik Sistem Pengaturan serta pernah menjuarai beberapa kompetisi dibidang karya tulis ilmiah dan pemrograman. E-mail: aulyiarasyid@yahoo.co.id